DONDERS
INSTITUTE

ISBN 978-94-6284-147-5

Radboud University Radboudumc

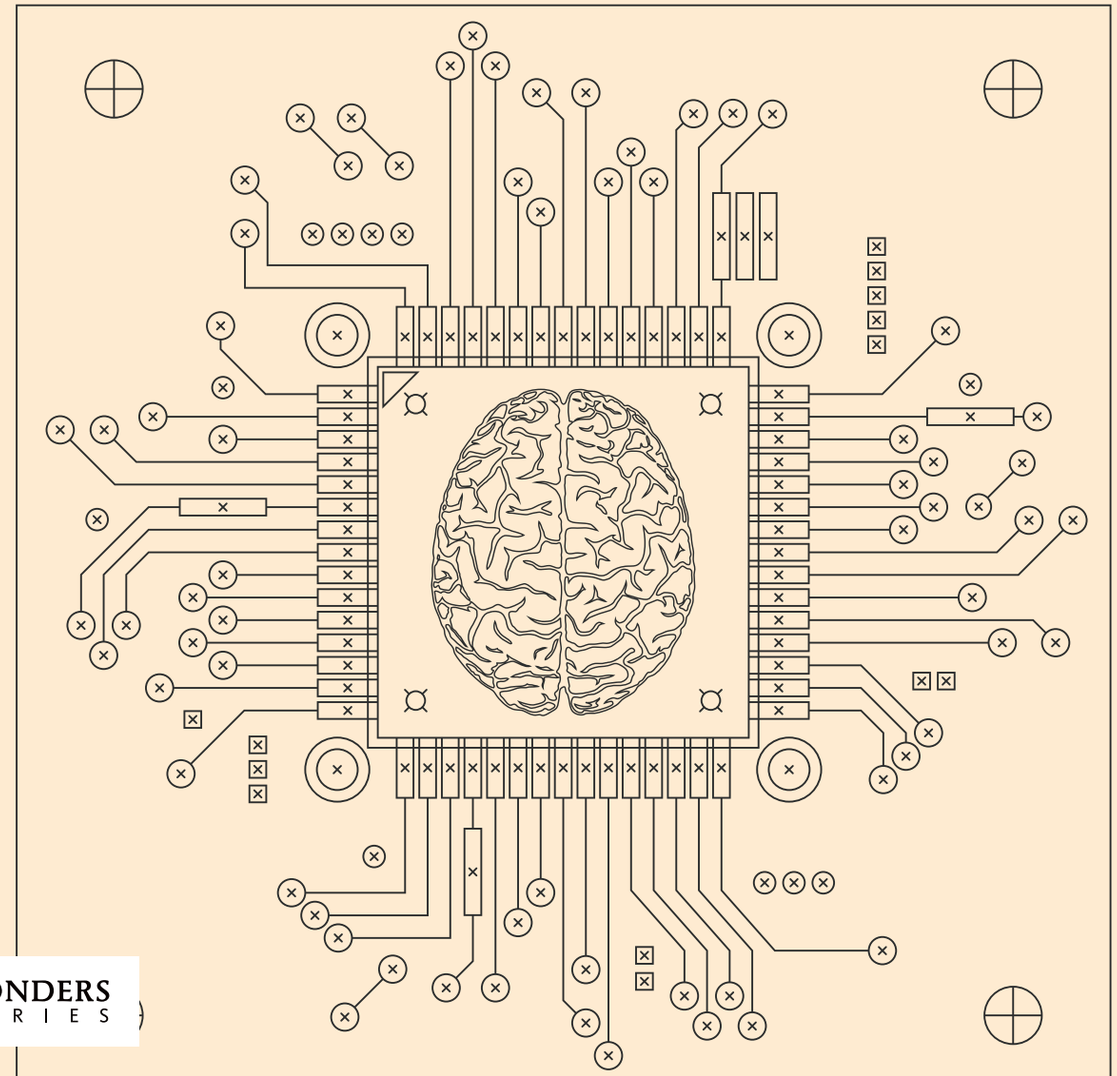
NEURAL CODING WITH DEEP LEARNING

Umut Güçlü

DONDERS SERIES 326

# NEURAL CODING WITH DEEP LEARNING

## UMUT GÜÇLÜ



DONDERS SERIES

# **Neural coding** with **deep learning**

**Umut Güçlü**

**ISBN**
978-94-6284-147-5

**Cover**
Robert Voight

# Neural coding with deep learning

## Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus
prof. dr. J.H.J.M. van Krieken,
volgens besluit van het college van decanen
in het openbaar te verdedigen op
donderdag 7 juni 2018 om
13:00 uur precies door


**Umut Güçlü**

geboren op
25 april 1986 te
Ankara (Turkije)

**Promotoren**
Prof. dr. ir. P.W.M. Desain
Prof. dr. D.G. Norris

**Copromotor**
Prof. dr. M.A.J. van Gerven

**Manuscriptcommissie**
Prof. dr. R.J.A. van Wezel
Prof. dr. P.R. Roelfsema (VU)
Dr. H.S. Scholte (UvA)

# Neural coding with deep learning

## Doctoral thesis

to obtain the degree of doctor
from Radboud University Nijmegen
on the authority of the Rector Magnificus
prof. dr. J.H.J.M. van Krieken,
according to the decision of the Council of Deans
to be defended in public on
Thursday, June 7, 2018 at
13:00 hours by


**Umut Güçlü**

Born on
April 25, 1986 in
Ankara (Turkey)

**Supervisors**
Prof. dr. ir. P.W.M. Desain
Prof. dr. D.G. Norris

**Cosupervisor**
Prof. dr. M.A.J. van Gerven

**Doctoral Thesis Committee**
Prof. dr. R.J.A. van Wezel
Prof. dr. P.R. Roelfsema (VU Amsterdam)
Dr. H.S. Scholte (University of Amsterdam)

> *The neurochemistry of the brain is astonishingly busy, the circuitry of a machine more wonderful than any devised by humans. But there is no evidence that its functioning is due to anything more than the $10^{14}$ neural connections that build an elegant architecture of consciousness.*

— **Carl Edward Sagan**
Cosmos (1980), p. 278

# Contents

# 1 Introduction

## 1.1 Introduction

A fundamental question in neuroscience is how the human brain makes sense of its environment. How does the brain manage to learn about, represent and recognize statistical invariances in the environment that guide its actions, ultimately ensuring our survival in a world that is in a continuous state of flux? The representation of invariant features of the environment is the subject matter of sensory neuroscience. In recent years, artificial neural networks have become a popular vehicle for probing how brains respond to their surroundings.

Artificial neural networks are computational models that consist of idealized artificial neurons and aim to mimic crucial aspects of information processing in biological neural networks. In engineering, they have been shown to be highly effective in complex problem-solving.

Artificial neural networks were initially conceived of as an approach to model mental or behavioral phenomena. This field is also referred to as *connectionism* (Hebb, 2002) and was popularized in the 1980's under the name *parallel distributed processing* (McClelland & Rumelhart, 1989). Artificial neural networks were inspired by their biological counterparts (Fukushima, 1980) but have since become tools that are mostly used by engineers. Interestingly, cognitive neuroscientists are now rediscovering the use of artificial neural networks in furthering our understanding of neural information processing in the human brain.

In this chapter, we review how artificial neural networks can be used to probe human brain function with a focus on the state-of-the-art results that emerged from this approach. We proceed as follows. First, we describe how one can model the mapping between stimuli and responses in the human brain through the development of *encoding models*. Next, we focus on how the

mapping from static naturalistic stimuli to neural responses can be realized using artificial neural networks. Then we move on to describing how brain responses induced by dynamically changing naturalistic environments can be modeled. We end this chapter by outlining the rest of this thesis.

## Modeling brain responses

We are interested in modeling how brains respond to their natural environment. That is, the goal is to model how complex and semantically rich naturalistic stimuli influence neural responses (Creutzfeldt & Nothdurft, 1978; Felsen & Dan, 2005). This objective can be achieved through the development of an *encoding model* which seeks to explain (1) how a stimulus modulates the activity of multiple neuronal populations and (2) how population activity affects data recorded at the sensor level (Kriegeskorte, 2015; Naselaris, Kay, Nishimoto & Gallant, 2011).

Consider an experiment in which $n$ (high-dimensional) stimuli $\mathbf{x}_t$ are presented to a subject at times $t_i$ with $i = 1, \ldots, N$. We use the $N \times K$ matrix

$$\mathbf{X} = [\mathbf{x}_{t_1}, \ldots, \mathbf{x}_{t_N}]^\top \tag{1.1}$$

to denote all $N$ stimuli of dimension $K$. For instance, $\mathbf{X}$ may be the sequence of all (vectorized) images that were shown in a vision experiment.

We are interested in the question how the external environment drives the responses of multiple neuronal populations to the stimuli $\mathbf{X}$. To this end, we introduce the notion of a *feature space*:

$$\phi(\mathbf{x}_t) = (\phi_1(\mathbf{x}_t), \ldots, \phi_P(\mathbf{x}_t))^\top \tag{1.2}$$

which captures sensory transformations.

During the experimental run, measurement vectors $\mathbf{y}$ are obtained across $Q$ sensors, reflecting the responses induced by the presented stimuli. For example, in functional magnetic resonance imaging (fMRI), $y_i$ is the blood oxygenation level-dependent (BOLD) response for voxel $i$ whereas in MEG it reflects the magnetic field generated by the (weighted) activity of multiple pools of neurons. Throughout the experiment, these measurement vectors are collected at times $u_j$ with $j = 1, \ldots, M$, yielding the $M \times Q$ matrix of measurements

$$\mathbf{Y} = [\mathbf{y}_{u_1}, \ldots, \mathbf{y}_{u_M}]^\top \tag{1.3}$$

An encoding model makes explicit how population activity is measured at the sensor level. These measurements may depend on the history of population activity, e.g. due to the hemodynamic lag when collecting fMRI BOLD data.

To accommodate for these lagged responses, let

$$\psi_t^{(i)} = (\phi_i(\mathbf{x}_{t-\Delta t}), \ldots, \phi_i(\mathbf{x}_t))^\top \tag{1.4}$$

denote the history of neural activity in the $i$th population for a given $\Delta t$. Let $\psi_t = \text{vec}(\psi_t^{(1)}, \ldots, \psi_t^{(P)})$. We now define the predicted response as

$$\widehat{\mathbf{y}}_t = \mathbf{r}(\psi_t) \tag{1.5}$$

with $\mathbf{r} = (r_1, \ldots, r_Q)^\top$ where $r_j$ is the *forward model* which maps (lagged) feature vectors to the $j$th sensor. Hence, development of an encoding model $\mathbf{r}(\psi_t)$ requires making a choice about the used feature representation as well as the used forward models.

As we will see in upcoming sections, artificial neural networks are an ideal basis for encoding models that map external stimuli to observed brain responses. Here, alternative network architectures provide neuroscientists with the freedom to incorporate different modeling assumptions. We refer to encoding models that employ artificial neural networks as *ANN-based encoding models*.

## Artificial neural networks

Before outlining how artificial neural networks can be used to model stimulus-response relationships in neuroscience, we provide the reader with some theoretical background.

Artificial neural networks (ANNs) are inspired by biological neural networks in two respects (Haykin, 1994). First, knowledge is acquired by the network through a learning process. Second, interneuron connection strengths referred to as (synaptic) weights are used to store the knowledge. Artificial neural networks have been around for over seventy years (McCulloch & Pitts, 1943; Copeland & Proudfoot, 1996) but have fallen in and out of favor several times throughout the course of their history. In the following, we describe the key elements of which neural networks are composed.

An *artificial neural network* (ANN) is a system of interconnected processing units (*artificial neurons*) which exchange messages between each other. An artificial neuron transforms a (vector-valued) input $\mathbf{x}$ into a scalar output $y$ by computing $y = f(a)$. Here, $f$ is the neuron's *activation function* and $a$ is known as the input *activation*, representing the neural firing rate. This activation is usually taken to be an inner product of the form $a = \mathbf{w}^T\mathbf{x}$, where $\mathbf{w}$ are adjustable parameters, also referred to as synaptic *weights* (an additional bias term can be absorbed in the weights by ensuring that one of the inputs is a constant). Each weight $w_i$ quantifies the strength with which the $i$th presynaptic input is connected to its post-synaptic neuron. The weights $\mathbf{w}$ can be tuned based on experience to maximize a certain objective function, thereby making ANNs capable of learning.

An ANN is fully characterized by the properties of its artificial neurons, its *architecture* (how its neurons are connected to one another), as well as the employed *learning algorithm*. The employed learning algorithm comes in three flavors: *supervised learning*, where the goal is to predict an output when given an input vector, *unsupervised learning*, where the goal is to discover a good internal representation of the input, and *reinforcement learning*, where the goal is to learn to select an action to maximize expected utility. For additional details, we refer to a number of excellent reviews (Schmidhuber, 2015; LeCun, Bengio & Hinton, 2015).

### Linear neural networks

Consider again our objective of modeling a stimulus-response mapping of the form shown in the previous equation. Let us assume that each measurement can be expressed as an instantaneous linear combination of input features (i.e. $\psi_t^{(i)} = \mathbf{x}_t$). That is, we assume that

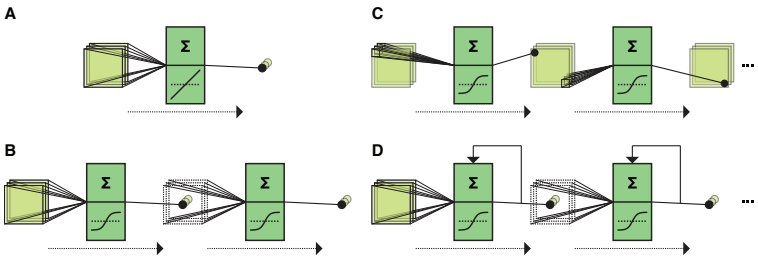



**Figure 1.1:** Artificial neural network architectures. ***A***: Linear neural network (multiple linear regression). ***B***: Multi-layer perceptron consisting of one hidden layer and non-linear activation functions. ***C***: Deep neural network with multiple convolutional hidden layers. ***D***: Recurrent neural network where hidden states feed back onto themselves.

$$\begin{aligned}\widehat{\mathbf{y}}_t =& (\mathbf{w}_1^\top \mathbf{x}_t, \ldots, \mathbf{w}_Q^\top \mathbf{x}_t)^\top && (1.6)\\ =& \mathbf{W}^\top \mathbf{x}_t && (1.7)\end{aligned}$$

with $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_Q]$ a $P \times Q$ matrix of adjustable parameters. It is easy to see that the previous equation implements a *linear neural network* with inputs $\mathbf{x}_t$, outputs $\widehat{\mathbf{y}}_t$, weights $\mathbf{W}$ and linear activation function $f(a) = a$ (see Figure 1.1A).

Training of this encoding model amounts to estimating the parameters $\mathbf{W}$. In the neural network community, estimation of the parameters is cast as a gradient descent problem. Let

$$\ell(\mathbf{w}) = \frac{1}{M} \sum_{t=1}^{M} ||\widehat{\mathbf{y}}_t - \mathbf{y}_t||^2 \tag{1.8}$$

denote the squared loss function. Let $\mathbf{w} = \text{vec}(\mathbf{W}) = (w_1, \ldots, w_K)$. Define the gradient

$$\nabla \ell = (\frac{\partial \ell}{\partial w_1}, \ldots, \frac{\partial \ell}{\partial w_K})^\top \tag{1.9}$$

By using the iteration

$$\mathbf{w}^{(n+1)} \leftarrow \mathbf{w}^{(n)} - \epsilon(\nabla \ell)_{\mathbf{w}^{(n)}} \tag{1.10}$$

with *learning rate* $\epsilon$, the weight vector converges to the optimal weight vector (Widrow & Hoff, 1960).

Linear neural networks (i.e., multiple linear regression) can be used as an encoding model for modeling the stimulus-response transformation. Such an encoding model can be used to predict stimulus-evoked responses directly. For example, they have previously been used for predicting human visual cortex voxel responses (measured with fMRI) to handwritten characters (Schoenmakers, Barth, Heskes & van Gerven, 2013).

### Multi-layer perceptrons

*Multi-layer perceptrons* (MLPs) are feed-forward neural networks whose artificial neurons are organized in terms of layers (see Figure 1.1B). The classical MLP consists of one layer of input neurons, one layer of hidden neurons, and one layer of output neurons. It computes a non-linear function of the inputs:

$$\mathbf{y} = \mathbf{g}(\mathbf{x}) \tag{1.11}$$

where, in case of the classical MLP, we have

$$\mathbf{g} = \mathbf{f}_2(\mathbf{W}_2^\top \mathbf{f}_1(\mathbf{W}_1^\top \mathbf{x})) \tag{1.12}$$

where the elements of $\mathbf{f}_i$ can be non-linear activation functions.

These properties extend the linear neural networks of the previous section. In fact, the *universal approximation theorem* tells us that MLPs consisting of one hidden layer can approximate any non-linear function with an arbitrary degree of precision given enough hidden neurons (Hornik, 1991; Cybenko, 1992).

In an MLP, minimization of a loss function also proceeds via a gradient descent procedure, as for the linear neural network case. However, due to the fact that the network consists of multiple layers, error derivatives need to be propagated backward from the output layer towards the input layer. It is this backpropagation algorithm which makes training of MLPs feasible (Rumelhart, Hinton & Williams, 1986).

Like linear neural networks, MLPs can also be used as an encoding model. For example, they have previously been used for predicting cat or monkey striate cortex neuron responses (measured with microelectrodes) to simple (e.g., bars) or compound (e.g., natural images) stimuli (Lehky, Sejnowski & Desimone, 1992; Lau, Stanley & Dan, 2002; Prenger, Wu, David & Gallant, 2004). The encoding models that comprise a fixed nonlinear feature space and a linear forward model can also be seen as MLPs, whose first layer weights are fixed and second layer weights are learned. Some examples thereof are the use of Gabor wavelets (Marĉelja, 1980) or semantic categories in combination with lasso or ridge regression to predict human visual cortex voxel responses (measured with fMRI) to natural images or movies (Kay, Naselaris, Prenger

& Gallant, 2008; Naselaris, Prenger, Kay, Oliver & Gallant, 2009; Nishimoto et al., 2011a).

### Deep neural networks

The classical MLP makes use of one hidden layer of artificial neurons. A recent trend is to train deep neural networks consisting of up to a thousand hidden layers (He, Zhang, Ren & Sun, 2015) (see Figure 1.1C).

Consider again the non-linear function $\mathbf{g}$. This function can also be written in terms of a composition of functions:

$$\mathbf{g}(\mathbf{x}) = (\phi_L \circ \cdots \circ \phi_1)(\mathbf{x}) \tag{1.13}$$

where $\phi_l$ is the transformation given by the artificial neurons that reside in the $l$-th layer of the neural network. In case the network contains more than one hidden layer, that is, $L > 2$, we speak of a *deep neural network* (DNN). Hence, DNNs are a special kind of MLP whose artificial neurons are organized in terms of layers.

Deep neural networks entered the stage about thirty five years ago with Fukushima's (1980) development of the *Neocognitron*. However, *deep learning*, i.e. backpropagation in deep neural networks has for a long time remained unfeasible, mainly due to instabilities in the weight updates. It was not until the start of the 21st century that deep learning gained traction. This can mainly be attributed to the curation of very large labeled datasets, the development of fast graphics processing units (GPUs), as well as the use of clever modifications to vanilla MLP training, such as *rectified linear activation functions*, *dropout learning* and *initialization methods* (LeCun et al., 2015).

This breakthrough in training of deep neural networks led to a truly Cambrian explosion of research in deep learning, leading to quantum leaps in e.g. object recognition (Krizhevsky, Sutskever & Hinton, 2012), natural language processing (Sutskever, Vinyals & Le, 2014) and reinforcement learning (Mnih et al., 2015; Silver et al., 2016), often matching and sometimes surpassing human-level performance.

One might argue that DNNs are not particularly useful given the universal approximation property of MLPs that consist of one hidden layer. However, it has been shown that many non-linear functions can be learned using much more compact deep architectures, compared to shallow architectures (Bengio, 2009). Moreover, the internal representations that emerge during DNN training have also been shown to be semantically meaningful (Zeiler & Fergus, 2013). That is, it was found that increasingly complex representations are learnt by DNNs that are trained, e.g., to predict the category to which an input image belongs.

Deep neural networks can be used as a feature space for modeling the feature-response transformation. Such a feature space can be combined with any forward model to predict stimulus-evoked responses. Similarly, DNNs can be used as a feature space for representational similarity analysis (RSA) (Kriegeskorte, 2008). Such a feature space can be compared to stimulus-evoked responses directly.

In one of the first studies on this topic, Yamins et al. (2014) predicted monkey V4 and IT neuron responses (measured with microelectrodes) to natural images with a DNN after training it for object recognition. They showed that intermediate and top layers of the DNN were highly predictive of monkey V4 and IT neuron responses to natural images, respectively. These results have later been successfully reproduced in the monkey visual cortex, and similar results have then been reported in the human visual cortex (Agrawal, Stansbury, Malik & Gallant, 2014; Khaligh-Razavi

& Kriegeskorte, 2014; Cadieu et al., 2014). Khaligh-Razavi and Kriegeskorte (2014) also compared (with RSA) the visual recognition performance of 37 different models including DNNs (Krizhevsky et al., 2012), Gabor wavelets (Marĉelja, 1980), GIST (Oliva & Torralba, 2001), HMAX (Poggio & Riesenhuber, 1999) and VisNet (Rolls & Milward, 2000) with one another as well as comparing the similarity of their representations to human and monkey IT representations (measured with fMRI). They showed that DNNs have not only the best visual recognition performance but also the most similar representations.

In Chapters 3-5, we extended these ideas to different tasks and areas. In Chapter 3, we probed human ventral stream representations (measured with fMRI) with a two-dimensional (spatial) DNN after it has been pretrained for object recognition (in photographs). In Chapter 4, we probed human dorsal stream representations (measured with fMRI) with a three-dimensional (spatiotemporal) convolutional neural networks after finetuning it for action recognition (in movies). In Chapter5, we moved beyond the visual cortex and to the auditory cortex. That is, we probed human superior temporal gyrus representations (measured with fMRI) with one-dimensional (spectral and/or temporal) convolutional neural networks after training them for music recognition (e.g., genre, instrument, mood, etc.). These studies showed the existence of a representational gradient such that increasingly deeper DNN layers better correspond to increasingly downstream areas in the human ventral stream, dorsal stream, and superior temporal gyrus as well as showing that this correspondence is driven by task-optimization and not exact architectural assumptions, which has been successfully reproduced since then (Seibert et al., 2016; Eickenberg, Gramfort, Varoquaux & Thirion, 2017). Similarly, Cichy, Khosla, Pantazis, Torralba and Oliva (2016), Seeliger et al. (2017) showed that the object recognition-optimized DNN-human ventral stream correspondence holds not only for space but also for time such that increasingly deep DNN layers better predict increasingly late stimulus-evoked human ventral stream

sensor or source responses (measured with MEG). Also, Kell, Yamins, Norman-Haignere and McDermott (2016) showed that the task-optimized DNN-human superior temporal cortex correspondence holds not only for music recognition- but also for speech recognition-optimized DNNs.

It is important to note that DNNs can also be used for classifying (Haxby, 2001; Kamitani & Tong, 2005), identifying (Kay et al., 2008; Mitchell et al., 2008) or reconstructing (Thirion et al., 2006; Miyawaki et al., 2008) a stimulus from voxel responses (measured with fMRI) in the human brain. They have previously been used for reconstructing perceived handwritten characters (van Gerven, de Lange & Heskes, 2010). In Chapters 3-4, we used them for identifying perceived natural images. They have more recently been used for classifying perceived and imagined natural images (Horikawa & Kamitani, 2017), and reconstructing perceived faces (Güçlütürk et al., 2017).

### Word embedding

Deep neural networks can be used to represent increasingly abstract stimulus features. Arguably, at the top of this hierarchy one may encounter conceptual representations. Such representations can also be captured more directly by focusing on linguistic input. We will now consider a special kind of MLP for learning word embeddings. Such word embeddings can be used to probe directly where conceptual knowledge is represented in the human brain.

When using linguistic stimuli (e.g. words), a neural network needs to be able to use individual words as input. One way to achieve this is by assuming that, given a vocabulary of $N$ words, the $n$th word is encoded as a one-hot vector of length $N$, consisting of all zeros except for a one at the $n$th index. A problem with this approach is that the number of words in a vocabulary can

run in the hundreds of thousands, making use of this sparse representation prohibitive in practice.

An alternative approach is to use a *word embedding* where each word is represented as a low-dimensional dense vector, providing a *distributed representation* for that word. The learning problem is to map the sparse high-dimensional representation to a dense low-dimensional representation (Bengio, Ducharme, Vincent & Janvin, 2003).

This learning problem can be cast in terms of an MLP. Given a sequence of words $w_1, w_2, \ldots, w_T$ that together make up a text, the *skip-gram model* maximizes the following objective function (Mikolov, Chen, Corrado & Dean, 2013b):

$$J = \frac{1}{T} \sum_{t=1}^{T} \sum_{j=-c}^{c} \log p(w_{t+j} \mid w_t) \tag{1.14}$$

Hence, the aim is to predict the context (surrounding) words given a target word.

This probability can be modeled using a neural network with one hidden layer (i.e. an MLP). The input-to-hidden weights are given by $\mathbf{U}$ and the hidden-to-output weights are given by $\mathbf{V}$. The probability of a context word $w'$ given a target word $w$ is then expressed as

$$p(w'|w) = \frac{\exp(\mathbf{v}_{w'}^{\top} \mathbf{u}_w)}{\sum_{i=1}^{W} \exp(\mathbf{v}_{w_i}^{\top} \mathbf{u}_w)} \tag{1.15}$$

where $\mathbf{u}_w$ and $\mathbf{v}_w$ are the input and output vectors associated with word $w$ and $W$ is the number of words in the vocabulary. The corresponding neural network uses a linear activation function for the hidden units and a softmax activation function for the output units. Let $\mathbf{e}(w)$ be the one-hot encoding of a word (e.g. $[0, 0, 0, 1, 0, 0, \ldots, 0, 0]$). Then, the word embedding of $w$ is given by $\text{vec}(w) \equiv \mathbf{e}(w)^T\mathbf{U} = \mathbf{u}_w = \mathbf{h}$.

Interestingly, the distributed representation of a word $\text{vec}(w)$ provides a semantically meaningful representation, even allowing for arithmetic expressions such as $\text{vec}(\textit{king}) - \text{vec}(\textit{man}) + \text{vec}(\textit{woman}) \approx \text{vec}(\textit{queen})$ (Mikolov, Yih & Zweig, 2013c).

The question of how the human brain encodes semantics has been extensively studied by mapping manually- or automatically-derived corpus representations to stimulus-evoked voxel responses (measured with fMRI) (Mitchell et al., 2008; Murphy, Talukdar & Mitchell, 2012; Huth, Nishimoto, Vu & Gallant, 2012; Fyshe, Murphy, Talukdar & Mitchell, 2013). Recent word embeddings such as W2V (Mikolov, Sutskever, Chen, Corrado & Dean, 2013a) and GloVe (Pennington, Socher & Manning, 2014) have been very successful in computational linguistics. They project words to continuous, distributed, low-dimensional vector space, where meanings of words, similarities between words and analogies are preserved.

Like DNNs, word embeddings can also be used as a feature space. Previously, Nishida, Gallant and Nishimoto (2015), Güçlü and van Gerven (2015) used word embeddings to predict downstream human visual cortex voxel responses (measured with fMRI) to semantic contents of natural images and movies. In Chapter 6, we used word embeddings while comparing different forward models. These studies showed that the human brain might be encoding semantics in a continuous, distributed, low-dimensional vector space, where many linguistic regularities are preserved. These results are reminiscent of the direct, predictive relationship

that was shown to exist between word co-occurrences and human brain activity (Mitchell et al., 2008).

### Unsupervised learning

So far, we have focused on models that were trained in a supervised manner. Another class of neural networks models is formed by those that are trained in an unsupervised manner on input data $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. Examples thereof are Hopfield networks (Hopfield, 1982), Boltzmann machines (Ackley, Hinton & Sejnowski, 1985) and deep belief networks (Hinton, Osindero & Teh, 2006). Rather than minimizing a loss function that measures the difference between observed and predicted output, these models aim to maximize the log probability of the input data, again using gradient descent procedures. That is, the update steps during gradient descent are given by

$$\Delta\theta = -\frac{\partial \sum_n \log p(\mathbf{x}_n)}{\partial\theta} \tag{1.16}$$

where $\theta$ is a model parameter.

Like supervised DNNs or word embeddings, unsupervised ANN variants can also be used as a feature space. Previously, van Gerven et al. (2010) used deep belief networks (Hinton et al., 2006) for reconstructing handwritten digits from stimulus-evoked voxel responses. Similarly, Güçlü and van Gerven (2013) used independent component analysis (Hyvärinen & Oja, 2000) for predicting voxel responses to handwritten digits and reconstructing them from stimulus-evoked voxel responses. In Chapter 2, we used sparse coding for predicting voxel responses to natural images and identifying them from stimulus-evoked voxel responses. All of these studies were concerned with early visual areas of the human

brain (whose responses were measured with fMRI) and showed that unsupervised ANN variants could account for low-level neural representations. Arguably, unsupervised learning offers a more biologically plausible explanation of neural representations in comparison to its supervised counterpart. However, unsupervised ANN variants have not been as successful in accounting for high-level neural representations (Khaligh-Razavi & Kriegeskorte, 2014).

### Recurrent neural networks

The feed-forward neural networks that have been reviewed so far are missing a key ingredient that is crucial to brain function, namely recurrence. Feed-forward neural networks make a new prediction at every time point, ignoring any temporal dependencies that might otherwise modulate their responses. However, it is clear that the brain does not function this way. That is, when confronted with a stimulus at a certain time point, the brain does not ignore everything that it has processed up to that time point. Rather, it takes into account the stimulus history and its responses are modulated by temporal dependencies.

In contrast to feed-forward neural networks, *recurrent neural networks* (RNNs) are implementations of dynamical systems that explicitly take temporal dependencies into account (Jordan, 1997; Elman, 1990) (see Figure 1.1D). Consider a RNN where inputs, hidden states and outputs at time $t$ are given by $\mathbf{x}_t$, $\mathbf{h}_t$ and $\mathbf{y}_t$, respectively. Effectively, RNNs can be seen as infinitely deep neural networks with the difference that each layer receives its own external input and produces an external output. Let $\mathbf{U}$ denote the input-to-hidden weights, $\mathbf{V}$ the hidden-to-hidden weights and $\mathbf{W}$ the hidden-to-output weights. We use $\mathbf{f}(\cdot)$ and $\mathbf{g}(\cdot)$ to denote the element-wise application of an activation function to a vector-valued input. In an RNN, updating of the hidden layers is given by

$\mathbf{h}_t = \mathbf{f}(\mathbf{U}\mathbf{x}_t + \mathbf{V}\mathbf{h}_{t-1})$ and updating of the output units is given by $\mathbf{y}_t = \mathbf{g}(\mathbf{W}\mathbf{h}_t)$.

A popular learning algorithm for recurrent neural networks is backpropagation through time (BPTT) (Werbos, 1990). It generalizes backpropagation for feed-forward networks to the recurrent case. This is done by unrolling the network so all cycles between units are removed while forcing the weights at each time point to be identical. In RNNs, when minimizing the error using gradient descent, one iterates over time rather than independent training examples, as in standard backpropagation.

It has been found that training of vanilla RNNs can be hard due to vanishing or exploding gradients in the BPPT gradient updates (Bengio, Simard & Frasconi, 1994). One way to improve RNN training is by endowing them with a memory, so events in the past can more easily update present network states. One way to realize this is through the use of *long short-term memory* (LSTM) layers (Hochreiter & Schmidhuber, 1997). LSTMs use memory cells surrounded by multiplicative gate units to store read, write and reset information. These gates, instead of sending their activities as inputs to other neurons, set the weights on edges connecting the rest of the neural net to the memory cell. LSTMs can be trained with backpropagation using somewhat more involved gradients.

Recurrent neural networks can be used as a forward model for modeling the feature-response transformation. Such a forward model can then be combined with any feature space to predict stimulus-evoked responses. Previously, Joukes, Hartmann and Krekelberg (2014) used RNNs to model the dynamics of the neurons in monkey MT (measured with microelectrodes) and showed that RNNs can reproduce properties of MT neurons in the monkey brain such as velocity computation better than nonrecurrent neural networks can. In Chapter 6, we used RNNs to model the dynamics of voxels in the human visual cortex (measured with

fMRI) and showed than RNNs can reproduce properties of visual cortex voxels in the human brain such as hemodynamic response better than nonrecurrent neural networks can.

## ANN-based encoding models

In summary, the way in which the ANN variants have been used in the literature and/or in this thesis can be grouped in the following three overlapping categories:

- Using linear neural networks (multiple linear regression) (Schoenmakers et al., 2013) or multi-layer perceptrons (Lehky et al., 1992; Lau et al., 2002; Prenger et al., 2004) for modeling the stimulus-response transformation (as an encoding model). Such an encoding model can be used to directly predict stimulus-evoked responses.

- Using deep neural networks (Yamins et al., 2014; Agrawal et al., 2014; Eickenberg et al., 2017), word embedding (Nishida et al., 2015; Güçlü & van Gerven, 2015) or unsupervised learning (van Gerven et al., 2010; Güçlü & van Gerven, 2013) for modeling the stimulus-feature transformation (as a feature space). Such a feature space can be combined with any forward model to indirectly predict stimulus-evoked responses. This is also the approach taken in Chapters 2-6.

- Using recurrent neural networks (Joukes et al., 2014) for modeling the feature-response transformation (as a forward model). Such a forward model can be combined with any feature space to indirectly predict stimulus-evoked responses. This is also the approach taken in Chapter 6.

## 1.2 Outline

**Chapter 2** In Chapter 2, to overcome the challenge of formalizing what stimulus features should modulate single voxel responses, we introduce a general approach for making directly testable predictions of single voxel responses to statistically adapted representations of ecologically valid stimuli. These representations are learned from unlabeled data without supervision. Our approach is validated using a parsimonious computational model of (i) how early visual cortical representations are adapted to statistical regularities in natural images and (ii) how populations of these representations are pooled by single voxels. This computational model is used to predict single voxel responses to natural images and identify natural images from stimulus-evoked multiple voxel responses. We show that statistically adapted low-level sparse and invariant representations of natural images better span the space of early visual cortical representations and can be more effectively exploited in stimulus identification than hand-designed Gabor wavelets. Our results demonstrate the potential of our approach to better probe unknown cortical representations.

**Chapter 3** In Chapter 3, we quantitatively show that there indeed exists an explicit gradient for feature complexity in the ventral pathway of the human brain. This is achieved by mapping thousands of stimulus features of increasing complexity across the cortical sheet using a deep neural network. Our approach also reveals a fine-grained functional specialization of downstream areas of the ventral stream. Furthermore, it allows decoding of representations from human brain activity at an unsurpassed degree of accuracy, confirming the quality of the developed approach. Stimulus features that successfully explain neural responses indicate that population receptive fields are explicitly tuned for object categorization. This provides strong support for the hypothesis that object categorization is a guiding principle in the functional organization of the primate ventral stream.

**Chapter 4** In Chapter 4, we explore whether deep neural networks also provide accurate predictions of neural responses across the dorsal visual pathway, which is thought to be devoted to motion processing and action recognition. This is achieved by training deep neural networks to recognize actions in videos and subsequently using them to predict neural responses while subjects are watching natural movies. Moreover, we explore whether dorsal stream representations are shared between subjects. In order to address this question, we examine if individual subject predictions can be made in a common representational space estimated via hyperalignment. Results show that a deep neural network trained for action recognition can be used to accurately predict how dorsal stream responds to natural movies, revealing a correspondence in representations of deep neural network layers and dorsal stream areas. It is also demonstrated that models operating in a common representational space can generalize to responses of multiple or even unseen individual subjects to novel spatio-temporal stimuli in both encoding and decoding settings, suggesting that a common representational space underlies dorsal stream responses across multiple subjects.

**Chapter 5** In Chapter 5, we develop task-optimized deep neural networks that achieve state-of-the-art performance in different evaluation scenarios for automatic music tagging. These deep neural networks are subsequently used to probe the neural representations of music. Representational similarity analysis reveal the existence of a representational gradient across the superior temporal gyrus. Anterior superior temporal gyrus is shown to be more sensitive to low-level stimulus features encoded in shallow deep neural network layers whereas posterior STG is shown to be more sensitive to high-level stimulus features encoded in deep deep neural network layers.

**Chapter 6** In Chapter 6, we investigate the extent to which recurrent neural network models can use their internal memories

for nonlinear processing of arbitrary feature sequences to predict feature-evoked response sequences as measured by functional magnetic resonance imaging. We show that the proposed recurrent neural network models can significantly outperform established response models by accurately estimating long-term dependencies that drive hemodynamic responses. The results open a new window into modeling the dynamics of brain activity in response to sensory stimuli.

# 2

# Unsupervised feature learning improves prediction of human brain activity in response to natural images

This chapter is based on Güçlü, U. and van Gerven, M. (2014). Unsupervised feature learning improves prediction of human brain activity in response to natural images. *PLOS Computational Biology*, 10(8):e1003724. https://doi.org/10.1371/journal.pcbi.1003724

## 2.1 Introduction

An important goal of contemporary cognitive neuroscience is to characterize the relationship between stimulus features and human brain activity. This relationship can be studied from two distinct but complementary perspectives of encoding and decoding (Dayan & Abbott, 2005). The encoding perspective is concerned with how certain aspects of the environment are stored in the brain and uses models that predict brain activity in response to certain stimulus features. Conversely, the decoding perspective uses models that predict specific stimulus features from stimulus-evoked brain activity and is concerned with how specific aspects of the environment are retrieved from the brain.

Stimulus-response relationships have been extensively studied in computational neuroscience to understand the information contained in individual or ensemble neuronal responses, based on different coding schemes (Brown, Kass & Mitra, 2004). The invasive nature of the measurement techniques of these studies has restricted human subjects to particular patient populations (Quiroga, Reddy, Kreiman, Koch & Fried, 2005; Pasley et al., 2012). However, with the advent of functional magnetic resonance imaging (fMRI), encoding and decoding in fMRI has made it possible to noninvasively characterize the relationship between stimulus features and human brain activity via localized changes in blood-oxygen-level dependent (BOLD) hemodynamic responses to sensory or cognitive stimulation (Naselaris et al., 2011).

Encoding models that predict single voxel responses to certain stimulus features typically comprise two main components. The first component is a (non)linear transformation from a stimulus space to a feature space. The second component is a (non)linear transformation from the feature space to a voxel space. Encoding models can be used to test alternative hypotheses about what a voxel represents since any encoding model embodies a specific

hypothesis about what stimulus features modulate the response of the voxel (Naselaris et al., 2011). Furthermore, encoding models can be converted to decoding models that predict specific stimulus features from stimulus-evoked multiple voxel responses. In particular, decoding models can be used to determine the specific class from which the stimulus was drawn (i.e. classification) (Haxby, 2001; Kamitani & Tong, 2005), identify the correct stimulus from a set of novel stimuli (i.e. identification) (Kay et al., 2008; Mitchell et al., 2008) or create a literal picture of the stimulus (i.e. reconstruction) (Thirion et al., 2006; Miyawaki et al., 2008; Schoenmakers et al., 2013).

The conventional approach to encoding and decoding makes use of feature spaces that are typically hand-designed by theorists or experimentalists (Kay et al., 2008; Mitchell et al., 2008; Miyawaki et al., 2008; Naselaris et al., 2009; Nishimoto et al., 2011a; Vu et al., 2011; Kay, Winawer, Rokem, Mezer & Wandell, 2013b). However, this approach is prone to the influence of subjective biases and restricted to a priori hypotheses. As a result, it severely restricts the scope of alternative hypotheses that can be formulated about what a voxel represents. This restriction is evident by a paucity of models that adequately characterize extrastriate visual cortical voxels.

A recent trend in models of visual population codes has been the adoption of natural images for the characterization of voxels that respond to visual stimulation (Kay et al., 2008; Naselaris et al., 2009). The motivation behind this trend is that natural images admit multiple feature spaces such as low-level edges, mid-level edge junctions, high-level object parts and complete objects that can modulate single voxel responses (Naselaris et al., 2011). Implicit about this motivation is the assumption that the brain is adapted to the statistical regularities in the environment (Barlow, 2012) such as those in natural images (Olshausen & Field, 1996; Bell & Sejnowski, 1997). At the same time, recent developments in theoretical neuroscience and machine learning have shown

that normative and predictive models of natural image statistics learn statistically adapted representations of natural images. As a result, they predict statistically adapted visual cortical representations, based on different coding principles. Some of these predictions have been shown to be similar to what is found in the primary visual cortex such as topographically organized simple and complex cell receptive fields (Hyvärinen, 2010).

Building on previous studies of visual population codes and natural image statistics, we introduce a general approach for making directly testable predictions of single voxel responses to statistically adapted representations of ecologically valid stimuli. To validate our approach, we use a parsimonious computational model that comprises two main components (Figure 2.1). The first component is a nonlinear feature model that transforms raw stimuli to stimulus features. In particular, the feature model learns the transformation from unlabeled data without supervision. The second component is a linear voxel model that transforms the stimulus features to voxel responses. We use an fMRI data set of voxel responses to natural images that were acquired from the early visual areas (i.e. V1, V2 and V3) of two subjects (i.e. S1 and S2) (Lescroart et al., 2011). We show that the encoding and decoding performance of this computational model is significantly better than that of a hand-designed Gabor wavelet pyramid (GWP) model of phase-invariant complex cells. The software that implements our approach is provided at http://www.ccnlab.net/research/.

## 2.2 Materials and methods

### Data

We used the fMRI data set (Lescroart et al., 2011) that was originally published in (Kay et al., 2008; Naselaris et al., 2009). Briefly,

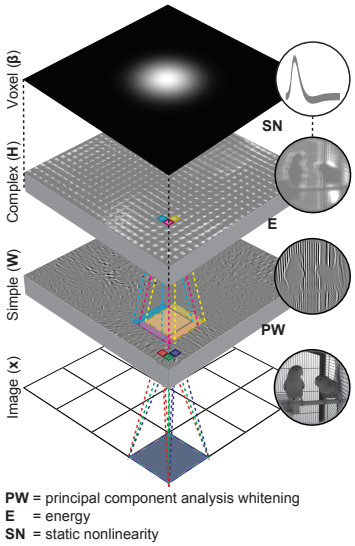



**Figure 2.1:** Encoding model. The encoding model predicts single voxel responses to images by nonlinearly transforming the images to complex cell responses and linearly transforming the complex cell responses to the single voxel responses. For example, the encoding model predicts a voxel response to a 128×128 image $\mathbf{x}$ as follows: Each of the 16 non-overlapping 32×32 patches of the image $\widehat{\mathbf{z}}^{(i)}$ is first vectorized, preprocessed and linearly transformed to 625 simple cell responses, i.e. $\mathbf{W}\mathbf{z}^{(i)}$ where $\mathbf{z}^{(i)}$ is a vectorized and preprocessed patch. Energies of the simple cells that are in each of the 625 partially overlapping 5×5 neighborhoods are then locally pooled, i.e. $\mathbf{H}(\mathbf{W}\mathbf{z}^{(i)})^2$, and nonlinearly transformed to one complex cell response, i.e. $\log(1 + \mathbf{H}(\mathbf{W}\mathbf{z}^{(i)})^2)$. Next, 10000 complex cell responses are linearly transformed to the voxel response, i.e. $\boldsymbol{\beta}^\top \phi(\mathbf{x})$ where $\phi(\mathbf{x}) = ((\log(1 + \mathbf{H}(\mathbf{W}\mathbf{z}^{(1)})^2)^\top, \ldots, (\log(1 + \mathbf{H}(\mathbf{W}\mathbf{z}^{(16)})^2)^\top)^\top$. The feature transformations are learned from unlabeled data. The voxel transformations are learned from feature-transformed stimulus-response pairs.

the data set contained 1750 and 120 stimulus-response pairs of two subjects (i.e. S1 and S2) in the estimation and validation sets, respectively. The stimulus-response pairs consisted of grayscale natural images of size 128×128 pixels and stimulus-evoked peak

BOLD hemodynamic responses of 5512 (S1) and 5275 (S2) voxels in the early visual areas (i.e. V1, V2 and V3). The details of the experimental procedures are presented in(Kay et al., 2008).

## Problem statement

### Encoding

Let $\mathbf{x} \in \mathbb{R}^d$ and $\mathbf{y} \in \mathbb{R}^q$ be a stimulus-response pair where $\mathbf{x}$ is a vector of pixels in a grayscale natural image, and $\mathbf{y}$ is a vector of voxel responses. Parameters $d$ and $q$ denote the number of pixels that make up the image and voxels, respectively. Given $\mathbf{x}$, we are interested in the problem of predicting $\mathbf{y}$:

$$\widehat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{y}|\phi(\mathbf{x})) = \mathbf{B}^\top \phi(\mathbf{x}) \tag{2.1}$$

where $\widehat{\mathbf{y}}$ is the predicted response to $\mathbf{x}$, and $p$ is the encoding distribution of $\mathbf{y}$ given $\phi(\mathbf{x})$. The function $\phi$ nonlinearly transforms $\mathbf{x}$ from the stimulus space to the feature space, and $\mathbf{B}$ linearly transforms $\phi(\mathbf{x})$ from the feature space to the voxel space.

### Decoding

Let $\mathbb{X}$ be a set of images that contains $\mathbf{x}$. Given $\mathbb{X}$ and $\mathbf{y}$, we are interested in the problem of identifying $\mathbf{x}$:

$$\widehat{\mathbf{x}} = \arg\max_{\mathbf{x} \in \mathbf{X}} \rho_{\mathbf{y}, \mathbf{B}^\top \phi(\mathbf{x})} \tag{2.2}$$

where $\widehat{\mathbf{x}}$ is the identified image from $\mathbf{y}$, and $\rho$ is the Pearson product-moment correlation coefficient between $\mathbf{y}$ and $\mathbf{B}^\top \phi(\mathbf{x})$.

Solving the encoding and decoding problems requires the definition and estimation of a feature model $\phi$ followed by a voxel model $\mathbf{B}$.

## Feature model

### Model definition

Following Hyvärinen and Hoyer (2001), we summarize the definition of the SC model. We start by defining a single-layer statistical generative model of whitened grayscale natural image patches. Assuming that a patch is generated by a linear superposition of latent variables that are non-Gaussian (in particular, sparse) and mutually independent, we first use independent component analysis to define the model by a linear transformation of independent components of the patch:

$$\mathbf{Z} = \mathbf{A}\mathbf{s} \tag{2.3}$$

where $\mathbf{z} \in \mathbb{R}^n$ is a vector of pixels in the patch, $\mathbf{A} \in \mathbb{R}^{n \times m}$ is a mixing matrix, and $\mathbf{s} \in \mathbb{R}^m$ is a vector of the components of $\mathbf{z}$ such that $m \leq n$. The parameters $n$ and $m$ denote the number of pixels and components, respectively. We then define $\mathbf{s}$ by inverting the linear system that is defined by $\mathbf{A}$:

$$\mathbf{s} = \mathbf{W}\mathbf{z} \tag{2.4}$$

where $\mathbf{W} \in \mathbb{R}^{m \times n}$ is an unmixing matrix such that $\mathbf{W} = \mathbf{A}^{-1}$. We constrain $\mathbf{W}$ to be orthonormal and $s_i$ to have unit variance such that $s_i$ are uncorrelated and unique, up to a multiplicative sign. Next, we define the joint probability of $\mathbf{s}$ by the product of the marginal probabilities of $s_i$ since $s_i$ are assumed to be independent:

$$p(\mathbf{s}) = \prod_{i=1}^{m} p(s_i) \tag{2.5}$$

where $p(s_i)$ are peaked at zero and have high kurtosis since $s_i$ are assumed to be sparse.

While one of the assumptions of the model is that $s_i$ are independent, their estimates are only maximally independent. As a result, residual dependencies remain between the estimates of $s_i$. We continue by modeling the nonlinear correlations of $s_i$ since $s_i$ are constrained to be linearly uncorrelated. In particular, we assume that the locally pooled energies of $s_i$ are sparse. Without loss of generality, we first arrange $s_i$ on a square grid graph that has circular boundary conditions. We then define the locally pooled energies of $s_i$ by the sum of the energies of $s_i$ that are in the same neighborhood:

$$\mathbf{c} = \mathbf{H}\mathbf{s}^2 \tag{2.6}$$

where $\mathbf{c} \in \mathbb{R}^m$ is a vector of the locally pooled energies of $s_i$ and $\mathbf{H} \in \mathbb{R}^{m \times m}$ is a neighborhood matrix such that $h_{i,j} = 1$ if $c_i$ pools the energy of $s_j$ and $h_{i,j} = 0$ otherwise. Next, we redefine $\log p(\mathbf{s})$ in terms of $\mathbf{c}$ to model both layers:

$$\log p(\mathbf{s}) \approx \sum_{i=1}^{m} G(c_i) \tag{2.7}$$

where $G$ is a convex function. Concretely, we use $G(c_i) = -\log(1+c_i)$.

In a neural interpretation, simple and complex cell responses can be defined as $\mathbf{s}$ and a static nonlinear function of $\mathbf{c}$, respectively. Concretely, we use $\log(1+\mathbf{c})$ to define the complex cell responses after we estimate the model.

### Model estimation

We use a modified gradient ascent method to estimate the model by maximizing the log-likelihood of $\mathbf{W}$ (equivalently, the sparseness of $\mathbf{c}$) given a set of patches:

$$\widehat{\mathbf{W}} = \arg\max_{\mathbf{W}} \mathcal{L}(\mathbf{W}|\mathbf{Z}) \tag{2.8}$$

where $\mathcal{L}(\mathbf{W}|\mathbf{Z}) = -\sum_{\mathbf{z}^{(i)}} \log p(\mathbf{H}(\mathbf{W}\mathbf{z}^{(i)})^2)$ is an approximation of the log-likelihood of $\mathbf{W}$ and $\mathbf{Z} = (\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \dots)$ is the set of patches. At each iteration, we first find the gradient of $\mathcal{L}(\mathbf{W}|\mathbf{Z})$:

$$\nabla_{\mathbf{W}} \mathcal{L}(\mathbf{W}|\mathbf{Z}) = -\mathbf{H}^\top (1 + \mathbf{H}(\mathbf{W}\mathbf{Z})^2)^{-1} \circ (2\mathbf{W}\mathbf{Z})\mathbf{Z}^\top \tag{2.9}$$

where $\circ$ is the Hadamard (element-wise) product. We then project it onto the tangent space of the constrained space (Edelman, Arias & Smith, 1998):

$$\overline{\nabla}_{\mathbf{W}}\mathcal{L}(\mathbf{W}|\mathbf{Z}) = \nabla_{\mathbf{W}}\mathcal{L}(\mathbf{W}|\mathbf{Z}) - \mathbf{W}\nabla_{\mathbf{W}}\mathcal{L}(\mathbf{W}|\mathbf{Z})^{\top}\mathbf{W} \tag{2.10}$$

Next, we use backtracking line search to choose a step size by reducing it geometrically with a rate from $(0, 1)$ until the Armijo-Goldstein condition holds (Boyd & Vandenberghe, 2004). Finally, we update $\mathbf{W}$ and find its nearest orthogonal matrix:

$$\mathbf{W} \leftarrow \mathbf{W} + \mu\overline{\nabla}_{\mathbf{W}}\mathcal{L}(\mathbf{W}|\mathbf{Z}) \tag{2.11}$$

$$\mathbf{W} \leftarrow (\mathbf{W}\mathbf{W}^{\top})^{-\frac{1}{2}}\mathbf{W} \tag{2.12}$$

where $\mu$ is the step size.

## Voxel model

### Model definition

We start by defining a model for each voxel. Assuming that $p(\mathbf{y}|\phi(\mathbf{x})) \sim \mathcal{N}(\mathbf{B}^{\top}\phi(\mathbf{x}), \mathbf{\Sigma})$, where $\mathbf{B} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_q) \in \mathbb{R}^{m\times q}$ and $\mathbf{\Sigma} = \text{diag}(\sigma_1^2, \ldots, \sigma_q^2) \in \mathbb{R}^{q\times q}$, we use linear regression to define the models by a weighted sum of $\phi(\mathbf{x})$:

$$y_i = \beta_i^{\top}\phi(\mathbf{x}) + \varepsilon_i \tag{2.13}$$

where $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$.

### Model estimation

We estimate the model using ridge regression:

$$\widehat{\beta}_i = \arg\min_{\beta_i} \frac{1}{N} \sum_{j=1}^{N} (y_i^{(j)} - \beta_i^\top \phi(\mathbf{x}^{(j)})) + \lambda_i ||\beta_i||_2^2 \tag{2.14}$$

where $\mathbf{X} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})^\top \in \mathbb{R}^{N \times d}$ and $\mathbf{Y} = (\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(N)})^\top \in \mathbb{R}^{N \times q}$ is an estimation set, and $\lambda_i \geq 0$ is a complexity parameter that controls the amount of regularization. The parameter $N$ denotes the number of stimulus-response pairs in the estimation set. We obtain $\widehat{\beta}_i$ as:

$$\widehat{\beta}_i = (\lambda_i \mathbf{I}_m + \mathbf{\Phi}^\top \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{Y}_i \tag{2.15}$$

where $\mathbf{\Phi} = (\phi(\mathbf{x}^{(1)}), \ldots, \phi(\mathbf{x}^{(N)}))^\top \in \mathbb{R}^{N \times m}$ and $\mathbf{Y}_i = (y_i^{(1)}, \ldots, y_i^{(N)})^\top \in \mathbb{R}^{N \times 1}$.

Since $m \gg N$, we solve the problem in a rotated coordinate system in which only the first $N$ coordinates of $\mathbf{\Phi}$ are nonzero (Hastie, Tibshirani & Friedman, 2009; Murphy et al., 2012). We first factorize $\mathbf{\Phi}$ using the singular value decomposition:

$$\mathbf{\Phi} = \mathbf{U}\mathbf{S}\mathbf{V}^\top \tag{2.16}$$

where $\mathbf{U}\mathbf{U}^\top = \mathbf{U}^\top\mathbf{U} = \mathbf{I}_N$, $\mathbf{S} = \text{diag}(\mathbf{s}) \in \mathbf{R}^{N\times N}$ and $\mathbf{V}^\top\mathbf{V} = \mathbf{I}_N$. The columns of $\mathbf{U}$, the diagonal entries of $\mathbf{S}$ and the columns of $\mathbf{V}$ are the left-singular vectors, the singular values and the right-singular vectors of $\mathbf{\Phi}$, respectively. We then reobtain $\widehat{\beta}_i$ as:

$$\widehat{\beta}_i = \mathbf{V}\text{diag}(\frac{\mathbf{s}}{\mathbf{s}\circ\mathbf{s}+\lambda_i})\mathbf{U}^\top\mathbf{Y}_i \tag{2.17}$$

where division is defined element-wise. The rotation reduces the complexity of the problem from $O(m^3)$ to $O(mN^2)$. To choose the optimal $\lambda_i$, we perform hyperparameter optimization using grid search guided by a generalized cross-validation approximation to leave-one-out cross-validation [60]. We define a grid by first sampling the effective degrees of freedom of the ridge regression fit from $[1, N]$ since its parameter space is bounded from above. The effective degrees of freedom of the ridge regression fit is defined as:

$$\text{df}(\lambda_i) = \sum_{j=1}^{N}\frac{s_j^2}{s_j^2+\lambda_i} \tag{2.18}$$

We then use Newton's method to solve $\mathbf{d}f$ for $\lambda_i$. Once the grid is defined, we choose the optimal that minimizes the generalized cross-validation error:

$$\widehat{\lambda}_i = \arg\min_{\lambda\in\mathbf{\Lambda}}\{\sum_{j=1}^{N}[\frac{y_i^{(j)}-\widehat{y}_i^{(j)}(\lambda)}{1-\mathbf{df}(\lambda)/N}]^2\} \tag{2.19}$$

where $\mathbf{\Lambda}$ is the grid, and $\widehat{y}_i^j(\lambda)$ is $\widehat{y}_i^{(j)}$ given a particular $\lambda$.

## Encoding and decoding

In the case of the SC model, each randomly sampled or non-overlapping patch was transformed to its principal components such that 625 components with the largest variance were retained and whitened prior to model estimation and validation. After the images were feature transformed, they were z-scored. The SC model of 625 simple and 625 complex cells was estimated from 50000 patches of size 32×32 pixels that were randomly sampled from the 1750 images of size 128×128 pixels in the estimation set. The details of the GWP model are presented in (Kay et al., 2008). The SC2 and GWP2 models were estimated from the 1750 feature-transformed stimulus-response pairs in the estimation set.

Voxel responses to an image of size 128×128 pixels were predicted as follows. In the case of the SC model, each 16 non-overlapping patch of size 32×32 pixels of the image were first transformed to the complex cell responses of the SC model (i.e. total of 625 complex cell responses per patch and 10000 complex cell responses per image). The 10000 complex cell responses of the SC model were then transformed to the voxel responses of the SC2 model. In the case of the GWP model, the image was first transformed to the complex cell responses of the GWP model (i.e. total of 10921 complex cell responses per image). The 10921 complex cell responses of the GWP model were then transformed to the voxel responses of the GWP2 model. The encoding performance was defined as the coefficient of determination between the observed and predicted voxel responses to the 120 images in the validation set across the two subjects.

A target image was identified from a set of candidate images as follows. Prior to identification, 500 voxels were selected without using the target image. The selected voxels were those whose responses were predicted best. The target image was identified as the candidate image such that the observed voxel responses to

the target image were most correlated with the predicted voxel responses to the candidate image (i.e. highest Pearson product-moment correlation coefficient between observed and predicted voxel responses). The decoding performance was defined as the accuracy of identifying the 120 images in the validation set from the set of 9264 candidate images. The set of candidate images contained the 120 images in the validation set and the 9144 images in the Caltech 101 data set (Fei-Fei, Fergus & Perona, 2007).

## 2.3 Results

### Feature models

To learn the feature transformation, we used a two-layer sparse coding (SC) model of 625 simple (i.e. first layer) and 625 complex (i.e. second layer) cells (Hyvärinen & Hoyer, 2001). Concretely, the simple cells were first arranged on a square grid graph that had circular boundary conditions. The weights between the simple and complex cells were then fixed such that each complex cell locally pooled the energies of 25 simple cells in a $5\times5$ neighborhood. There were a total of 625 partially overlapping neighborhoods that were centered around the 625 simple cells. Next, the weights between the input and the simple cells were estimated from 50000 patches of size $32\times32$ pixels by maximizing the sparseness of the locally pooled simple cell energies. Each simple cell was fully connected to the input (i.e. patch of size $32\times32$ pixels). The patches were randomly sampled from the 1750 images of size $128\times128$ pixels in the estimation set. To maximize the sparseness, the energy function (i.e. square nonlinearity) encourages the simple cell responses to be similar within the neighborhoods while the sparsity function (i.e. convex nonlinearity) encourages the locally pooled simple cell energies to be thinly dispersed across the neighborhoods. As a result, the simple cells that are in the same

neighborhood have simultaneous activation and similar preferred parameters. Since the neighborhoods overlap, the preferred parameters of the simple and complex cells change smoothly across the grid graph. Finally, the complex cell responses of the SC model were defined as a static nonlinear function of the locally pooled simple cell energies after model estimation (i.e. total of 625 complex cell responses per patch of size 32×32 pixels and 10000 complex cell responses per image of size 128×128 pixels). The SC model learned topographically organized, spatially localized, oriented and bandpass simple and complex cell receptive fields that were similar to those found in the primary visual cortex (Figure 2.2*A*) (Hubel & Wiesel, 1968; Valois, Albrecht & Thorell, 1982; Jones & Palmer, 1987; Parker & Hawken, 1988).

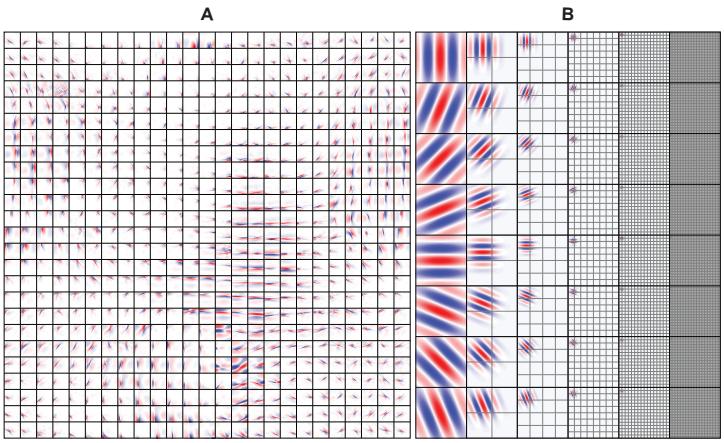



**Figure 2.2:** Simple cell receptive fields. ***A***: Simple cell receptive fields of the SC model. Each square is of size 32×32 pixels and shows the inverse weights between the input and a simple cell. The receptive fields were topographically organized, spatially localized, oriented and bandpass, similar to those found in the primary visual cortex. ***B***: Simple cell receptive fields of the GWP model. Each square is of size 128×128 pixels and shows an even-symmetric Gabor wavelet. The grids show the locations of the remaining Gabor wavelets that were used. The receptive fields spanned eight orientations and six spatial frequencies.

To establish a baseline, we used a GWP model (Daugman, 1985; Jones & Palmer, 1987; Lee, 1996) of 10921 phase-invariant complex cells (Kay et al., 2008). Variants of this model were used in a series of seminal encoding and decoding studies (Kay et al., 2008; Naselaris et al., 2009; Nishimoto et al., 2011a; Kay et al., 2013b). Note that the fMRI data set was the same as that in (Kay et al., 2008; Naselaris et al., 2009). Concretely, the GWP model was a hand-designed population of quadrature-phase Gabor wavelets that spanned a range of locations, orientations and spatial frequencies (Figure 2.2*B*). Each wavelet was fully connected to the input (i.e. image of size $128 \times 128$ pixels). The complex cell responses of the GWP model were defined as a static nonlinear function of the pooled energies of the quadrature-phase wavelets that had the same location, orientation and spatial frequency (i.e. total of 10921 complex cell responses per image of size $128 \times 128$ pixels).

## Voxel models

To learn the voxel transformation, we used regularized linear regression. The voxel models were estimated from the 1750 feature-transformed stimulus-response pairs in the estimation set by minimizing the $L^2$ penalized least squares loss function. The combination of a voxel model with the complex cells of the SC and GWP models resulted in two encoding models (i.e. SC2 and GWP2 models). The SC2 model linearly pooled the 10000 complex cell responses of the SC model. The GWP2 model linearly pooled the 10921 complex cell responses of the GWP model.

## Receptive fields

We first analyzed the receptive fields of the SC model (i.e. simple and complex cell receptive fields). The preferred phase, location, orientation and spatial frequency of the simple and complex

cells were quantified as the corresponding parameters of Gabor wavelets that were fit to their receptive fields. The preferred parameter maps of the simple and complex cells were constructed by arranging their preferred parameters on the grid graph (Figure 2.3). Most adjacent simple and complex cells had similar location, orientation and spatial frequency preference, whereas they had different phase preference. In agreement with Hyvärinen and Hoyer (2001), the preferred phase, location and orientation maps reproduced some of the salient features of the columnar organization of the primary visual cortex such as lack of spatial structure (DeAngelis, Ghose, Ohzawa & Freeman, 1999), retinotopy (Hubel & Wiesel, 1977) and pinwheels (Blasdel, 1992), respectively. In contrast to Hyvärinen and Hoyer (2001), the preferred spatial frequency maps failed to reproduce cytochrome oxidase blobs (Tootell, Silverman, Hamilton, Switkes & Valois, 1988). The preferred phase map of the simple cells suggests that the complex cells are more invariant to phase and location than the simple cells since the complex cells pooled the energies of the simple cells that had different phase preference. To verify the invariance that is suggested by the preferred phase map of the simple cells, the population parameter tuning curves of the simple and complex cells were constructed by fitting Gaussian functions to the median of their responses to Gabor wavelets that had different parameters (Figure 2.4). Like the simple cells, most complex cells were selective to orientation (i.e. standard deviation of 21.8° versus 22.9°) and spatial frequency (i.e. standard deviation of 0.52 versus 0.54 in normalized units). Unlike the simple cells, most complex cells were more invariant to phase (i.e. standard deviation of 50.0° versus 158.1°) and location (i.e. standard deviation of 3.70 pixels versus 5.86 pixels). Therefore, they optimally responded to Gabor wavelets that had a specific orientation and spatial frequency, regardless of their phase and exact position.

We then analyzed the receptive fields of the SC2 model (i.e. voxel receptive fields). The eccentricity and size of the recept-

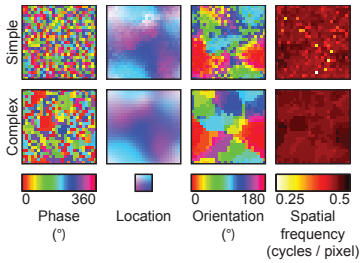



**Figure 2.3:** Preferred parameter maps of the SC model. The phase, location, orientation and spatial frequency preference of the simple and complex cells were quantified as the corresponding parameters of Gabor wavelets that were fit to their receptive fields. Each pixel in a parameter map shows the corresponding preferred parameter of a simple or complex cell. The adjacent simple and complex cells had similar location, orientation and spatial frequency preference but different phase preference.

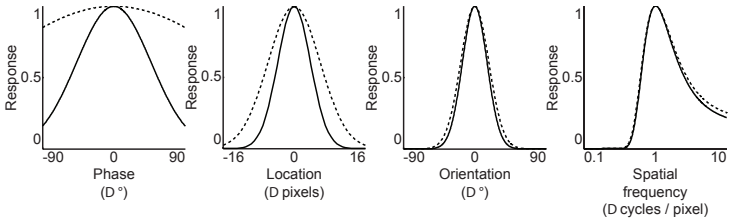



**Figure 2.4:** Population parameter tuning curves of the SC model. The population phase, location, orientation and spatial frequency tunings of the simple (solid lines) and complex cells (dashed lines) were quantified by fitting Gaussian functions to the median of their responses to Gabor wavelets that had different parameters. Each curve shows the median of their responses as a function of change in their preferred parameter. The complex cells were more invariant to phase and location than the simple cells.

ive fields were quantified as the mean and standard deviation of two-dimensional Gaussian functions that were fit to the voxel responses to point stimuli at different locations, respectively. The orientation and spatial frequency tuning of the receptive fields were taken to be the voxel responses to sine-wave gratings that spanned a range of orientations and spatial frequencies. While

the eccentricity, size and orientation tuning varied across voxels, most voxels were tuned to relatively high spatial frequencies (Figure 2.5*A* and Figure 2.5*B*). The mean predicted voxel responses to sine-wave gratings that had oblique orientations were higher than those that had cardinal orientations and this difference decreased with spatial frequency (Figure 2.5*C*). While this result is in contrast to those of the majority of previous single-unit recording and fMRI studies (Mansfield, 1974; Furmanski & Engel, 2000), it is in agreement with those of Swisher et al. (2010). In line withDumoulin and Wandell, 2008; Smith, 2001, the receptive field size systematically increased from V1 to V3 and from low receptive field eccentricity to high receptive field eccentricity (Figure 2.6). The properties of the GWP2 model were similar to those in (Kay et al., 2008). The relationship between the receptive field parameters (i.e. size, eccentricity, area) of the GWP2 model were the same as those of the SC2 model. However, the GWP2 model did not have a large orientation bias.

## Encoding

The encoding performance of the SC2 and GWP2 models was defined as the coefficient of determination ($R^2$) between the observed and predicted voxel responses to the 120 images in the validation set across the two subjects. The performance of the SC2 model was found to be significantly higher than that of the GWP2 model (binomial test, $p \ll 0.05$). Figures 2.7*A* and 2.7*B* compare the performance of the models across the voxels that survived an $R^2$ threshold of 0.1. The mean $R^2$ of the SC2 model systematically decreased from 0.28 across 28% of the voxels in V1 to 0.21 across 11% of the voxels in V3. In contrast, the mean $R^2$ of the GWP2 model systematically decreased from 0.24 across 24% of the voxels in V1 to 0.16 across 6% of the voxels in V3. Figure 2.7*C* compares the performance of the models in each voxel. More than 71% of the voxels that did not survive the threshold in each area and more than 92% of the voxels that survived the

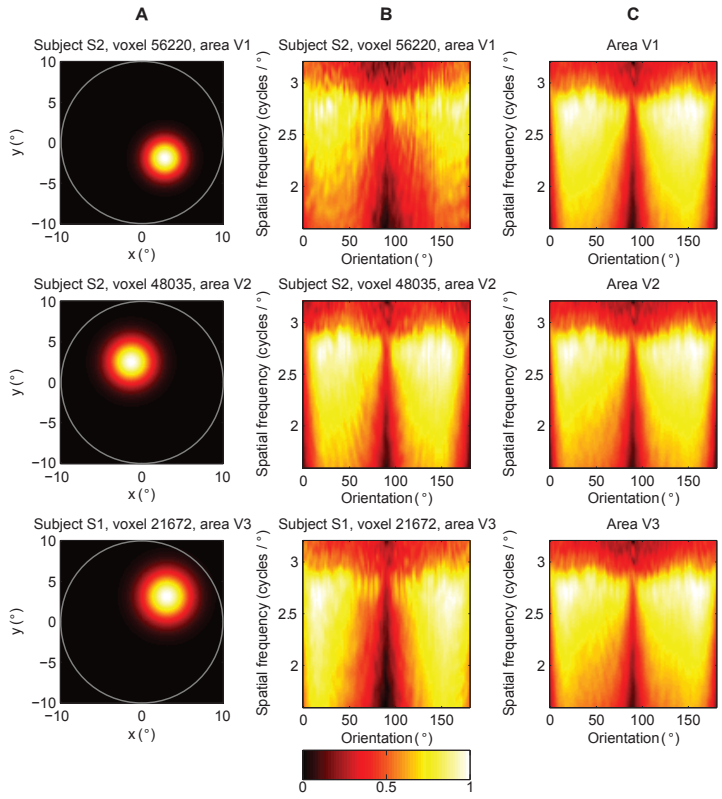



**Figure 2.5:** Receptive fields of the SC2 model. The parameter tuning varied across the voxels and had a bias for high spatial frequencies and oblique orientations. ***A***: Two-dimensional Gaussian functions that were fit to the responses of three representative voxels to point stimuli at different locations. ***B***: Responses of three representative voxels to sine-wave gratings that spanned a range of orientations and spatial frequencies. ***C***: Mean responses across the voxels to sine-wave gratings that spanned a range of orientations and spatial frequencies.

threshold in each area were better predicted by the SC2 model than the GWP2 model. These results suggest that statistically adapted low-level sparse representations of natural images better span the space of early visual cortical representations than the Gabor wavelets.

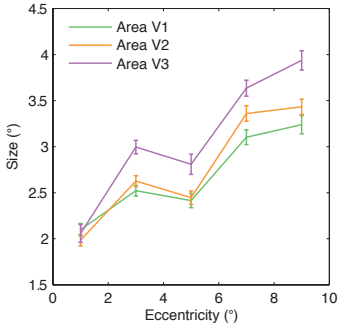



**Figure 2.6:** Receptive field size of the SC2 model as a function of receptive field eccentricity of the SC2 model and area. The eccentricity and size of the receptive fields were quantified as the mean and standard deviation of two-dimensional Gaussian functions that were fit to the voxel responses to point stimuli at different locations, respectively. The receptive field size systematically increased from low to high receptive field eccentricity and from area V1 to V3. Error bars show $\pm 1$ SEM across the voxels (bootstrapping method).

## Decoding

The decoding performance of the SC2 and GWP2 models was defined as the accuracy of identifying the 120 images in the validation set from a set of 9264 candidate images. The set of candidate images contained the 120 images in the validation set and the 9144 images in the Caltech 101 data set (Fei-Fei et al., 2007). Note that the set of candidate images was ten- to hundred-fold larger than the sets in (Kay et al., 2008) but comparable to the largest set in (Vu et al., 2011). The performance of the SC2 model was found to be significantly higher than that of the GWP2 model (binomial test, $p < 0.05$). Figure 2.8 compares the performance of the models. The mean accuracy of the SC2 model across the subjects was 61%. In contrast, the mean accuracy of the GWP2 model across the subjects was 49%. The chance-level accuracy was 0.01%. These results suggest that statistically adapted low-level sparse representations of natural images can

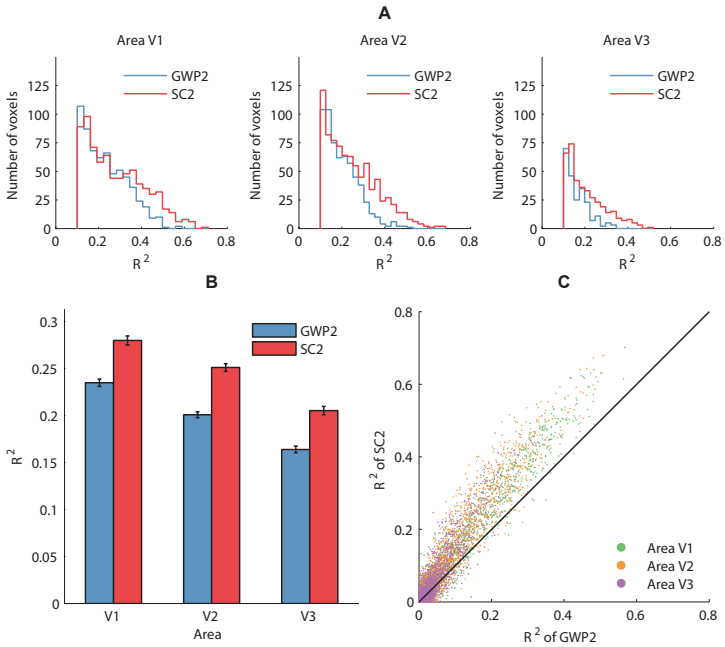



**Figure 2.7:** Encoding performance of the SC2 and GWP2 models. The encoding performance was defined as $R^2$ between the observed and predicted voxel responses to the 120 images in the validation set across the two subjects. The encoding performance of the SC2 model was significantly higher than that of the GWP2 model. ***A***: Prediction $R^2$ across the voxels that survived the $R^2$ threshold of 0.1. ***B***: Mean prediction $R^2$ across the voxels that survived the $R^2$ threshold of 0.1. Error bars show $\pm 1$ SEM across the voxels (bootstrapping method). ***C***: Prediction $R^2$ in each voxel.

be more effectively exploited in stimulus identification than the Gabor wavelets.

## Spatial invariance

In principle, the SC2 and GWP2 models should have some degree of spatial invariance since they linearly pooled the responses of the complex cells that displayed insensitivity to local stimulus position. Spatial invariance is of particular importance for decoding since a

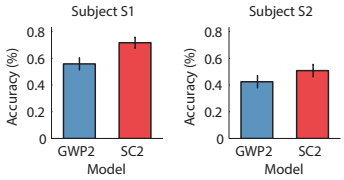



**Figure 2.8:** Decoding performance of the SC2 and GWP2 models. The decoding performance was defined as the accuracy of identifying the 120 images in the validation set from a set of 9264 candidate images. The decoding performance of the SC2 model was significantly higher than that of the GWP2 model. Error bars show $\pm 1$ SEM across the images in the validation set (bootstrapping method). A more detailed figure that shows the identified images is provided at http://www.ccnlab.net/research/.

reliable decoder should be able to identify a stimulus, regardless of its exact position. Furthermore, a difference between the degree of spatial invariance of the models can be a contributing factor to the difference between their performance. To analyze the spatial invariance of the models, we evaluated their encoding and decoding performance after translating the images in the validation set by 0.8° (i.e. approximately the standard deviation of the population location tuning curves of the complex cells of the SC model) in a random dimension (Figure 2.9). The encoding and decoding performance of the models was found to decrease after the translations. Unlike the encoding performance of the GWP2 model, that of the SC2 model decreased less in V3 than V1. This result suggests greater spatial invariance in V3 than V1. The difference between the mean $R^2$ of the models across the voxels that survived the threshold before the translations increased from 0.05 to 0.11. The difference between the mean accuracy of the models across the subjects increased from 12% to 24%. These results suggest that the SC2 model is more tolerant to local translations in stimulus position than the GWP2 model.

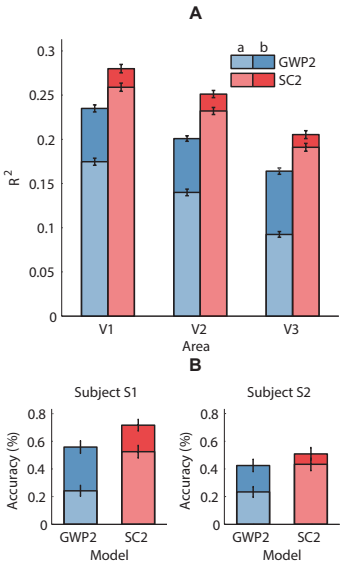



**Figure 2.9:** Mean prediction $R^2$ and identification accuracy of the SC2 and GWP2 models after (a) and before (b) translating the images in the validation set by 0.8° in a random dimension. The SC2 model was more invariant than the GWP2 model and its invariance increased from V1 to V3. ***A***: Mean prediction $R^2$ across the voxels that survived the $R^2$ threshold of 0.1 in the case of (b). Error bars show $\pm 1$ SEM across the voxels (bootstrapping method). ***B***: Identification accuracy. Error bars show $\pm 1$ SEM across the images in the validation set (bootstrapping method).

## Control models

Since the SC2 and GWP2 models had different nonlinearities (i.e. pooling and static nonlinearity), a direct evaluation of the contribution of their components (i.e. representations and nonlinearities) to the difference between their encoding performance was not possible. Therefore, we estimated two control models that pooled the same static nonlinear function of the simple cell responses of the SC and GWP models. The static nonlinear function was a compressive nonlinearity (i.e. $\log(1 + |s|)$ where $s$ is a simple cell response). The compressive nonlinearity roughly

accounts for insensitivities by increasing responses to a stimulus that is not entirely within a receptive field (Kay, Winawer, Mezer & Wandell, 2013a). The simple cell responses were defined as the linear responses of the first layer of the SC model and the even-symmetric Gabor wavelets. While the performance of the compressive nonlinear SC model was significantly higher than that of the compressive nonlinear GWP model, the difference between the performance of the compressive nonlinear models was significantly lower than that of the SC2 and GWP2 models (Figure 2.10). This result suggests that both the representations and the nonlinearities of the SC2 model contribute to the difference between the encoding performance of the SC2 and GWP2 models.

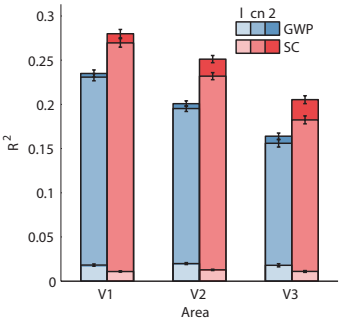



**Figure 2.10:** Mean prediction $R^2$ of the linear one-layer (l), compressive nonlinear one-layer (cn) and nonlinear two-layer (2) SC and GWP models across the voxels that survived the $R^2$ threshold of 0.1 in the case of (2). The mean prediction $R^2$ of the linear one-layer models were below the $R^2$ threshold of 0.1. The mean prediction $R^2$ of the nonlinear SC models were significantly better than those of the nonlinear GWP models. The compressive nonlinearity and the nonlinear second layer increased the mean prediction $R^2$ of the linear and compressive nonlinear models, respectively. The nonlinear second layer increased the mean prediction $R^2$ of the compressive nonlinear SC model more than it increased that of the compressive nonlinear GWP model. The error bars show $\pm 1$ SEM across the voxels (bootstrapping method).

To verify the contribution of the nonlinearities to the individual encoding performance of the SC2 and GWP2 models, we estimated two more control models that pooled a linear function of the simple cell responses of the SC and GWP models. We used linear models since they retain selectivities that are discarded by nonlinearities. We found that the performance of the linear models were significantly lower than that of the compressive nonlinear, SC2 and GWP2 models (Figure 2.10). This result confirms the contribution of the nonlinearities that introduced the insensitivities to the individual encoding performance of the SC2 and GWP2 models.

## 2.4 Discussion

This study addresses the question of how to model feature spaces to better predict brain activity. We introduced a general approach for making directly testable predictions of single voxel responses to statistically adapted representations of ecologically valid stimuli. Our approach relies on unsupervised learning of a feature model followed by supervised learning of a voxel model. To benchmark our approach against the conventional approach that makes use of predefined feature spaces, we compared a two-layer sparse coding model of simple and complex cells with a Gabor wavelet pyramid model of phase-invariant complex cells. While the GWP model is the fundamental building block of many state-of-the-art encoding and decoding models, the GWP2 model was found to be significantly outperformed by the SC2 model. We used control models to determine the contribution of the different components of the SC2 and GWP2 models to this performance difference. Analyses revealed that the SC2 model better accounts for both the representations and the nonlinearities of the voxels in the early visual areas than the GWP2 model. Given that the representations of the SC2 model are qualitatively similar to those of the GWP model, their contribution to this performance difference suggests

that the SC model automatically learns an optimal set of spatially localized, oriented and bandpass representations that better span the space of early visual cortical representations since it adapts to the same statistical regularities in the environment as the brain is assumed to be adapted to (Hyvärinen, 2010).

Our approach eliminates the need for predefining feature spaces. However, the SC model does have a number of free parameters (e.g. patch size, number of simple and complex cells, etc.) that must either be specified by hand or using model selection methods such as cross-validation. Because of computational considerations, we used the same free parameters as those in (Hyvärinen & Hoyer, 2001). While the choice of these free parameters can influence what the SC model can learn, the SC2 model was shown to outperform the GWP2 model even without cross-validation. Next to cross-validation, other methods that also infer these free parameters can further improve the performance of the SC2 model. One method is to first estimate voxel receptive fields using any approach and then use these estimates as free parameters (e.g. voxel receptive field eccentricity as patch size) of voxel-specific feature models. Another method is to use more sophisticated nonparametric Bayesian sparse factor models (Knowles & Ghahramani, 2011) that can simultaneously learn sparse representations while inferring their number. Furthermore, our approach included only feedforward projections such that representations and responses were solely determined by stimuli. However, taking top-down modulatory effects into account is essential to adequately characterize how sensory information is represented and processed in the brain. For example, attention has been shown to warp semantic representations across the human brain (Çukur, Nishimoto, Huth & Gallant, 2013), and prior expectations have been shown to bias sensory representations in visual cortex (Kok, Brouwer, van Gerven & de Lange, 2013). Extensions of our approach that include feedback projections can be used to address the question of how representations and responses are influenced by top-down processes.

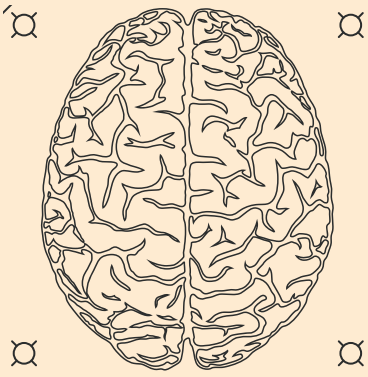
Further extensions of our approach can be used to probe mid- to high-level extrastriate visual cortical representations in a fully automated manner. In particular, the SC model can be replaced by highly nonlinear multi-layer statistical models of natural images that learn hierarchical feature spaces (i.e. deep learning (Bengio, Courville & Vincent, 2013)). Some of the feature spaces that are learned by these models such as mid-level edge junctions have been shown to match well with neural response functions in area V2 (Lee, Ekanadham & Ng, 2007). Models that learn even higher-level representations such as high-level object parts (Lee, Grosse, Ranganath & Ng, 2009) or complete objects (Le, 2013) can be used to probe extrastriate visual cortical representations. For example, heterogenous hierarchical convolutional neural networks have been shown to predict the representational dissimilarity matrices that characterize representations in human inferior temporal gyrus (Yamins et al., 2014). Similar models have been shown to learn feature spaces that are admitted by stimulus sets other than natural images, both within the visual modality (e.g. natural movies (Le, Zou, Yeung & Ng, 2011)) as well as in other modalities (e.g. auditory or somatosensory (Saxe, Bhand, Mudur, Suresh & Ng, 2011)). These models can be used to probe cortical representations in different sensory modalities.

One approach to estimate deep models is to maximize the likelihood of all layers at the same time. However, this approach is not scalable and requires the computation of intractable partition functions that are impossible to integrate analytically and computationally expensive to integrate numerically. Nevertheless, methods such as score-matching (Hyvärinen, 2005) and noise-contrastive estimation (Gutmann & Hyvärinen, 2012) have been used to estimate unnormalized nonlinear multi-layer statistical models of natural images (Köster & Hyvärinen, 2010; Gutmann & Hyvärinen, 2013). An alternative approach is to use models such as deep belief networks that comprise multiple layers of restricted Boltzmann machines. These models can be scaled by convolution (Lee et al., 2009) and estimated by maximizing the

likelihood of one layer at a time, using the output of each layer as input for the subsequent layer (Hinton et al., 2006). Importantly, generative models such as deep belief networks make it possible to sample stimuli based on internal network states. Conditioning these internal network states on stimulus-evoked brain activity results in a generative approach to decoding. For example, we have previously shown that a deep belief network that comprise multiple layers of conditional restricted Boltzmann machines can reconstruct handwritten digits by sampling from the model after conditioning it on stimulus-evoked multiple voxel responses (van Gerven et al., 2010).

While introducing a new approach to probe cortical representations, this study complements other developments in encoding and decoding. For example, encoding models that involve computations to account for contrast saturation or heterogeneous contrast energy were shown to improve prediction of single voxel responses to visual stimuli (Kay et al., 2013b). At the same time, these modeling efforts go hand in hand with developments in fMRI such as the improvements in contrast-to-noise ratio and spatial resolution that are facilitated by increases in magnetic field strength (Duyn, 2012). For example, spatial features of orientation-selective columns in humans were demonstrated by using high-field fMRI (Yacoub, Harel & Ugurbil, 2008). Jointly, such developments can provide novel insights into how cortical representations are learned, encoded and transformed.

In conclusion, we introduced a general approach that improves prediction of human brain activity in response to natural images. Our approach primarily relies on unsupervised learning of transformations of raw stimuli to representations that span the space of cortical representations. These representations can also be effectively exploited in stimulus classification, identification or reconstruction. Taken together, unsupervised feature learning heralds new ways to characterize the relationship between stimulus features and human brain activity.



# 3

# Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream

This chapter is based on: Güçlü, U. and van Gerven, M. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *The Journal of Neuroscience*, 35(27):10005–10014. https://doi.org/10.1523/JNEUROSCI.5023-14.2015

## 3.1 Introduction

Human beings are extremely adept at recognizing complex objects based on elementary visual sensations. Object recognition appears to be solved in the primate brain via a cascade of neural computations along the visual ventral stream that represents increasingly complex stimulus features, which derive from the retinal input (Tanaka, 1996). That is, neurons in early visual areas have smaller receptive fields (RFs) and respond to simple features such as edge orientation (Hubel & Wiesel, 1962), whereas neurons further along the ventral pathway have larger RFs and are more invariant to transformations and can be selective for complex shapes (Gross, Rocha-Miranda & Bender, 1972; Hung, 2005). Despite a consensus concerning a steady progression in feature complexity, it remains nontrivial to quantify such a progression across multiple regions in the human ventral stream. Furthermore, while the RFs in early visual area V1 have been characterized in terms of preferred orientation, location, and spatial frequency (Jones & Palmer, 1987), exactly what stimulus features are represented in downstream areas is less clear (Cox, 2014).

To probe how stimulus features of varying complexity are mapped across the cortical sheet, we made use of a feedforward deep neural network (DNN), which was trained to predict the object category of over a million natural images. DNNs consist of multiple layers where deeper layers can be shown to respond to increasingly complex stimulus features (Zeiler & Fergus, 2013). We used the representations that emerge after training a DNN to predict BOLD responses to complex naturalistic stimuli. We show that this framework yields state-of-the-art encoding and decoding performances, improving on results from earlier studies that used nonlinear feature models as the basis for neural encoding and decoding (Kay et al., 2008; van Gerven et al., 2010; Güçlü & van Gerven, 2014).

Predictions were made in progressively downstream areas of the ventral stream, moving from striate area V1 along extrastriate areas V2 and V4, all the way up to downstream area LO. Individual neural network layers were used to predict single-voxel responses to natural images. This allowed us to isolate different voxel groups, whose population RFs (pRFs) (Dumoulin & Wandell, 2008) are best predicted by a particular neural network layer. Using this approach, we were able to determine how RF properties, such as complexity, invariance, and size, correlate with the position of voxels in the visual hierarchy.

Next, by using individual features in the neural network to predict voxel responses, we were able to map how individual low-, mid-, and high-level stimulus features are represented across the ventral stream. This mapping procedure provides detailed insight into how stimulus features are represented across cortex and indicates that particular visual areas show a fine-grained functional specialization. Our results show that DNNs accurately predict neural responses to naturalistic stimuli and suggest that object categorization is a guiding principle for the formation of receptive field properties in ventral stream.

## 3.2 Materials and methods

### Experimental data

To examine the functional organization of the ventral stream, we reanalyzed the dataset that was originally published in (Kay et al., 2008) and (Naselaris et al., 2009). Hence, the experimental design, MRI acquisition protocol, and preprocessing of the data are identical to those described in these studies. Here, we restrict ourselves to a brief overview of the details already presented in those studies.

For each of two male subjects (S1 and S2), five sessions of data were collected as subjects were presented with natural images. Training and test data were collected in the same scan sessions. The total number of images used for training and testing were 1750 and 120, respectively. Each training image was repeated two times, and each test image was repeated 13 times.

Stimuli consisted of grayscale natural images ($20 \times 20°$) drawn randomly from different photographic collections. Subjects fixated on a central white square ($0.2 \times 0.2°$). Stimuli were flashed at 200 ms intervals for 1 s followed by 3 s of gray background in successive 4 s trials.

Data were acquired using a 4 T INOVA MR scanner and a quadrature transmit/receive surface coil. Eighteen coronal slices were acquired covering occipital cortex (slice thickness 2.25 mm, slice gap 0.25 mm, field of view $128 \times 128$ mm$^2$). fMRI data were acquired using a gradient-echo EPI pulse sequence (matrix size $64 \times 64$, TR 1 s, TE 28 ms, flip angle 20°, spatial resolution $2 \times 2 \times 2.5$ mm$^3$).

fMRI scans were coregistered and used to estimate voxel-specific response time courses. After deconvolution of these time courses from the time series data, an estimate of response amplitude was obtained for each presented unique image in each voxel. Voxels were assigned to visual areas using retinotopic mapping data acquired in separate sessions. Additionally, anatomical and functional volumes were coregistered manually. Surface reconstruction and flattening were performed using FreeSurfer software (http://surfer.nmr.mgh.harvard.edu).

## Encoding model

To transform images to BOLD responses, we developed an encoding model consisting of two components, as shown in Figure 3.1.

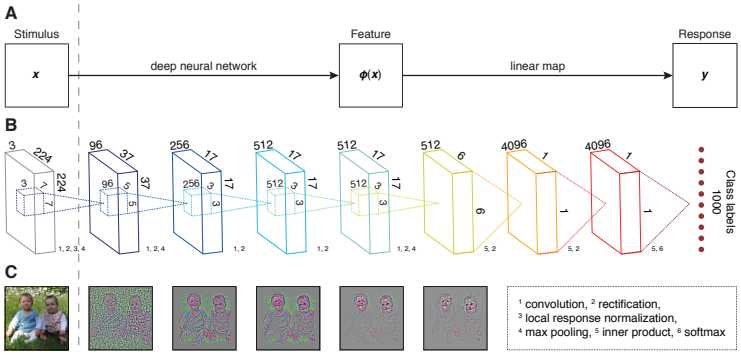



**Figure 3.1:** DNN-based encoding framework. ***A***: Schematic of the encoding model that transforms a visual stimulus to a voxel response in two stages. First, a deep (convolutional) neural network transforms the visual stimulus ($\boldsymbol{x}$) to multiple layers of feature representations. Then, a linear mapping transforms a layer of feature representations to a voxel response ($\boldsymbol{y}$). ***B***: Schematic of the deep neural network where each layer of artificial neurons uses one or more of the following (non)linear transformations: convolution, rectification, local response normalization, max pooling, inner product, and softmax. ***C***: Reconstruction of an example image from the activities in the first five layers.

The first component of the encoding model is a feature model that transforms a visual stimulus to a nonlinear feature representation. To this end, we used the pretrained CNN-S architecture of Chatfield, Simonyan, Vedaldi and Zisserman (2014) as a feature model. This architecture is similar to that of Krizhevsky et al. (2012) and consists of five convolutional and three fully connected layers of artificial neurons. Each artificial neuron in the convolutional layers corresponds to a feature detector that is replicated over spatial locations, which we refer to as a feature map. That is, a representation of a stimulus feature across space. In contrast, each artificial neuron in the fully connected layers took all features at all locations in the previous layer as its input. The artificial neurons used rectified linear activation functions in Layers 1–7. A softmax function was used in Layer 8 to transform feature activations to class labels. Layer 1 additionally used local response normalization, implementing lateralized inhibition

between feature maps at the same spatial position. Finally, Layers 1, 2, and 5 used max pooling, which can be interpreted as a form of nonlinear downsampling that introduces invariances to small translations of the input.

The DNN was trained on ~1.2 million augmented (by random crops, horizontal mirroring, and color jittering) natural images that are each labeled as 1 of 1000 object categories. The natural images were taken from the ImageNet (2012) dataset (Deng et al., 2009). Each input image was represented as a 224 × 224 matrix for each of three RGB color channels. The Caffe framework (Jia et al., 2014) was used to train the DNN with stochastic gradient descent using momentum and weight decay. The learning rate was initialized to 0.001 and decreased by a factor of 10 when the validation error stopped decreasing. Dropout regularization was applied to Layers 6 and 7 of the DNN (Hinton, Srivastava, Krizhevsky, Sutskever & Salakhutdinov, 2012).

The second component of the encoding model is a linear response model that transforms nonlinear feature representations to a voxel response. A separate response model was trained for each voxel using regularized linear regression. The used estimation procedure was described in detail previously (Güçlü & van Gerven, 2014). To examine which DNN layer was most predictive of individual voxel responses, we used each one of the eight layers of feature representations as input. Additionally, to investigate how individual features are represented across the cortical surface, we trained separate response models for each feature map/voxel combination. After estimation of the regression coefficients $\beta_i$, we obtain $\mu_i(\boldsymbol{x}) = \beta_i{}^T \phi(\boldsymbol{x})$ as the predicted response of voxel $i$ to input stimulus $\boldsymbol{x}$ given a chosen feature representation $\phi(\boldsymbol{x})$. Voxel response models were estimated using the entire training set and evaluated on the test set.

## Quantification of model performance

To quantify how well the nonlinear feature representations predict voxel responses, we define a voxel's prediction accuracy as the Pearson's correlation coefficient ($r$) between its observed and predicted responses on the test set. For a group of voxels, the median $r$ was used to express its prediction accuracy. To account for performance variability across voxels, we compared prediction accuracies of voxels with their SNRs and the mean activities of the DNN layers across the training set. SNR was estimated as the ratio between the mean time series and the median of the absolute differences between the successive time points in the detrended time series of the voxels. Next to computing the prediction accuracy for individual voxels, we can use the accuracy of reconstructing a presented image from observed brain activity as a measure of model performance. Let $\boldsymbol{X}$ be a set of candidate stimuli that contains the target stimulus. Given the response $\boldsymbol{y}$ to the target stimulus, we can compute the most probable stimulus by maximizing the likelihood: $\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\boldsymbol{X}} \{-(\boldsymbol{y} - \mu(\boldsymbol{x}))^T \Sigma^{-1} (\boldsymbol{y} - \mu(\boldsymbol{x}))\}$ where $\mu(\boldsymbol{x})$ is the predicted response by the encoding model using the optimal layer assignment for each voxel, and $\Sigma$ is an estimate of the noise covariance. Concretely, a target stimulus is identified from a set of potential stimuli as follows. First, those voxels that have the highest prediction accuracy on the test set are chosen without using the target stimulus. The target stimulus is identified as the potential stimulus that has the highest likelihood. The identification accuracy is defined as the percentage of 120 stimuli in the test set that are correctly identified from the set of 1870 (training and test set) potential stimuli. To further improve decoding performance, predictions were made by refitting an encoding model for each voxel. Each of these encoding models took as input all features in the preferred layer of its corresponding voxel at the locations that fall within its estimated receptive field. The receptive field of each voxel was estimated by refitting another set of encoding models that take as input all features in the preferred layer of the voxel at individual

spatial locations. The receptive field was then taken as the spatial locations whose corresponding models accurately predicted the response of the voxel.

## Control models

To further assess the performance of our DNN approach, we compared it with a number of control models. First, to establish a baseline, we used a Gabor wavelet pyramid (GWP) basis as a nonlinear feature representation, as this has been shown to produce state-of-the-art results on the same dataset (Kay et al., 2008). Concretely, the GWP model is a hand-designed population of quadrature-phase Gabor wavelets that have different locations, orientations, and spatial frequencies. The responses of the GWP model are defined as the square root of the pooled energies of the quadrature-phase wavelets that have the same location, orientation, and spatial frequency. Our GWP model is similar to that in (Kay et al., 2008) except that it operates on $256 \times 256$ pixel images rather than $128 \times 128$ pixel images.

Second, to examine to what extent our results depend on particular architectural assumptions, we compared the encoding performance of the DNN with that of nine different pretrained DNNs. Concretely, we used the DNNs that are colloquially referred to as vgg-verydeep-16 and vgg-verydeep-19 (Simonyan & Zisserman, 2014); vgg-f, vgg-m, vgg-m-2048, vgg-m-1024, and vgg-m-128 (Chatfield et al., 2014); caffe-ref (Jia et al., 2014); and caffe-alex (Krizhevsky et al., 2012). These DNNs differ in their exact architectures (number of layers, number of artificial neurons in a layer, number and type of pooling and local response normalization, size of receptive fields, etc.). However, they have been trained on the same dataset (i.e., ImageNet) for the same task (i.e., object categorization). Two of these DNNs have more than five convolutional layers (i.e., vgg-verydeep-16 and vgg-verydeep-19). To enable layer-wise comparison, we grouped the convolutional

layers of these DNNs to have five groups and used the outputs of the last layer in a group as the outputs of the entire group.

Third, to test whether results are explained by optimizing the DNN for categorization, we compared its encoding performance with that of nine random DNNs that share the same architecture, but whose weights are drawn from a zero mean and unit variance multivariate Gaussian. Note that in the case of random DNNs, only the feature models have Gaussian parameters, but the parameters of the response models are still estimated from the training set. We quantified the prediction accuracies and layer assignments of a set of nine (pretrained or random) DNNs as the median of the prediction accuracies and layer assignments of the DNNs in the set, respectively. Comparison of two models was performed on the held-out test set across the combination of all significant voxels of both model and subject (that were selected using cross-validation on the training set) for each individual visual area separately.

## Analysis of internal representations

A deconvolutional network (Zeiler & Fergus, 2013) was used to reconstruct the internal representations of artificial neurons as follows. The image that maximally activates each artificial neuron was selected from the ImageNet (2012) validation set. The image was first forward propagated through the network until it reached the layer of the neuron of interest. Then all the activations except the maximum activation of the neuron were set to zero. Finally, the activation of the neuron was deconvolved to produce a representation in image space. In this setting, deconvolution is defined as inverting the order of the layers, transposing the filters, and replacing max pooling with max unpooling.

After an initial evaluation of the internal representations, nine feature classes were defined such that they were representative of the most common low-level (blob, contrast, and edge), mid-level

(contour, shape, and texture), and high-level (irregular pattern and object part and entire object) internal representations of the 1888 artificial neurons in the convolutional layers. To further characterize the internal representations, each of these neurons was assigned a predefined label by a naive subject across five hour-long sessions. The subject was presented with four instantiations of the internal representations of the neurons (together with the images that were used to reconstruct them) in a random order and was asked to assign one of the following feature classes: blob, contrast, edge, contour, shape, texture, irregular pattern, object part and entire object. Each instantiation corresponded to the reconstruction of the internal representation of a neuron using one of the four images that activated the neuron the most.

## Analysis of voxel groups

Individual voxels were assigned to their optimal layer according to maximal prediction accuracy computed using fivefold cross-validation on the training data. Subsequently, voxels were grouped together according to their assigned neural network layer. Voxel group properties were estimated as follows. The RF center of a voxel is defined as the location on the feature map that has the greatest regression coefficient. The RF size, complexity, and invariance of the *k*th voxel group are taken to be those of the *k*th neural network layer. Layer size is defined as the size of the internal representations of the artificial neurons in the layer. Layer complexity is defined as the mean Kolmogorov complexity (K) of the internal representations of the artificial neurons in that layer, approximated by their normalized compressed file size. Layer invariance is defined as the median full-width at half-maximum of two-dimensional Gaussian surfaces that have been fitted to the two-dimensional response surfaces of the artificial neurons in that layer (reflecting tolerance to small translations of a stimulus feature). The two-dimensional response surface of an artificial neuron is estimated as follows. First, the reconstruction of the

internal representation of the artificial neuron is shifted to different spatial locations. Next, the activity of the neuron is computed for each translation and a two-dimensional response surface is constructed.

## Clustering of voxel responses

To identify fine-grained structure within individual visual areas, we made use of hyperalignment (Haxby et al., 2011) followed by nonparametric Bayesian biclustering (Meeds & Roweis, 2007). Hyperalignment was used to transform the individual functional data of the two subjects to a common representational space. Concretely, the individual representational space of the subject that has the most number of voxels was selected as the initial common representational space. The common representational space was then iteratively updated for 100 iterations. In each iteration, a Procrustes transformation was used to project the individual functional data of the two subjects to the common representational space, after which the common representational space was set to the mean of the individual functional data of the two subjects. Each visual area was hyperaligned separately. Nonparametric Bayesian biclustering was used to simultaneously cluster rows and columns of a $z$-scored prediction accuracy matrix where rows and columns correspond to individual feature maps and region-specific voxels of the common representational space, respectively. This allows for a fine-grained analysis of representational structure present within individual visual areas. Our approach assumes that the observed prediction accuracies for each feature map/voxel pair are drawn from a Gaussian with zero mean and unit standard deviation. A collapsed Gibbs sampler was used to generate samples from the posterior of cluster assignments over feature maps and voxels (https://github.com/ppletscher/npbb). The Gibbs sampler was run for 30 iterations and the cluster assignment produced by the final iteration was used as our estimate of cluster structure.

## 3.3 Results

### Deep neural networks accurately capture voxel responses across the ventral stream

We used fivefold cross-validation to assign voxels to one of the eight layers of the DNN. Each voxel was assigned to the layer of the DNN that resulted in the lowest cross-validation error on the training set. Those voxels whose prediction accuracy was not significantly better than chance were discarded ($p > 5e\text{-}8$ for both subjects, Bonferroni corrected for number of layers and voxels, Student's $t$ test across cross-validated training images within subjects), leaving 3381 of 25,915 voxels for S1 and 1185 of 26,329 voxels for S2. If we consider only the main afferent pathway of the ventral stream (V1, V2, V4, and LO) then 1786 of 6017 and 768 of 4875 voxels remained for S1 and S2, respectively.

The nonlinear feature representations allowed accurate prediction of voxel responses in different visual areas (Figure 3.2*A*). The prediction accuracy of the V1, V2, V4, and LO voxels was 0.51, 0.46, 0.30, and 0.30 for S1 and 0.42, 0.38, 0.26, and 0.29 for S2 (Figure 3.2*B*). Prediction accuracy was significantly correlated with voxel SNR (Figure 3.2*C*; $r = 0.27$ and $p = 2e\text{-}308$ for S1; $r = 0.22$ and $p = 1e\text{-}286$ for S2; Student's $t$ test across voxels within subjects) and the mean activity of the neural network layers ($r = 0.93$ and $p = 0.0028$ for S1; $r = 0.89$ and $p = 0.0078$ for S2; Student's $t$ test across voxel groups within subjects) over the training set, providing a partial explanation for the difference in the prediction accuracy of the low- and high-level voxels.

Given the high accuracy with which individual voxel responses can be predicted, it is natural to ask to what extent the deep model allows decoding of a perceived stimulus from observed multiple voxel responses alone. To answer this question, we eval-

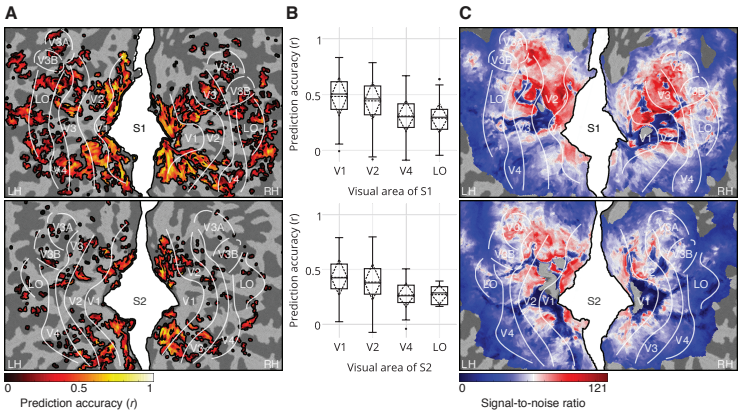



**Figure 3.2:** The DNN model accurately predicts voxel responses across the occipital cortex. ***A***: Prediction accuracies of the significant voxels across the occipital cortex ($p$ < 2e-6 for both subjects, Bonferroni corrected for number of voxels, Student's $t$ test across cross-validated training images within subjects). ***B***: Prediction accuracies of the significant voxels across V1, V2, V4, and LO ($p$ < 5e-8 for both subjects, Bonferroni corrected for number of layers and voxels, Student's $t$ test across cross-validated training images within subjects). ***C***: SNRs of the voxels across the occipital cortex.

uated three decoding models: striate (V1), an extrastriate (V2, V4, LO, and beyond), and a ventral stream (striate and extrastriate). All decoding models performed significantly better than the chance level of 5e-4% ($p$ < 2e-308 for all decoding models and subjects, binomial test across test images within subjects). Given observed voxel responses, the striate decoding model correctly identified a stimulus from a set of 1870 potential stimuli at 96 (S1; 500 voxels) and 79% (S2; 250 voxels) accuracy, whereas the extrastriate decoding model correctly identified a stimulus from the same set at 95 (S1; 500 voxels) and 63% (S2; 250 voxels) accuracy. This result suggests that a combination of the striate and extrastriate decoding models would have a higher accuracy since the striate voxels can be used to resolve the ambiguities in the feature representations of the extrastriate voxels and vice versa. As expected, the ventral stream decoding model showed higher

identification accuracy than either of the previous two decoding models. It identified the correct stimulus from a set of 1870 potential stimuli at 98 (S1; 1000 voxels) and 93% (S2; 500 voxels) accuracy. This improves on earlier approaches that exclusively used low-level features (Kay et al., 2008; Güçlü & van Gerven, 2014), demonstrating that mid- and high-level features are also important for identification.

## Image decoding is driven by discriminative and categorical information

To examine to what extent decoding performance is driven by discrimination (identifying an image based on its unique characteristics) versus categorization (identifying an image based on categorical information), the following analysis was performed. We manually assigned each image in the test set to one of two categories (animate vs inanimate), as this appears to be the strongest categorical division in inferior temporal cortex (Khaligh-Razavi & Kriegeskorte, 2014). A total of 99 of 120 test images could be assigned to either of these categories and were used for further analysis. Subsequently, we computed the pairwise linear correlations between the observed and predicted responses to each pair of images. The correlations were computed separately for low-level (V1), mid-level (V2 and V4), and high-level (LO and beyond) voxels. It was found that the correlation between the observed and predicted responses to an image was significantly higher than the mean correlation between the observed responses to the same image and the predicted responses to different images, regardless of their category ($p < 5e\text{-}13$ for both subjects, Bonferroni corrected for number of conditions, Student's $t$ test across test images within subjects). This points toward identification based on each image's unique characteristics. For high-level voxels only, it was additionally found that the mean pairwise correlation between the observed and predicted responses to a pair of same category images was significantly higher than that of different category

images ($p$ < 7e-25 for both subjects, Bonferroni corrected for number of conditions, Student's $t$ test across test images within subjects). This indicates that for downstream areas, not only unique characteristics of an image, but also its semantic content is involved in response prediction.

## Voxel groups exhibit coherent representational characteristics

We pooled voxels that were assigned to the same DNN layer together and analyzed their properties. The responses of successive voxel groups were more partially correlated than those of nonsuccessive voxel groups (Figure 3.3*A*). This shows that information flow mainly takes place between neighboring visual areas, providing quantitative evidence for the thesis that the visual ventral stream is hierarchically organized (Markov et al., 2013), with downstream areas processing increasingly complex features of the retinal input.

The voxel RFs in each group covered almost the entire field of view, with more voxels dedicated to foveal than peripheral vision (Figure 3.3*B*). While there was some degree of overlap between the internal representations of the successive voxel groups, results of the behavioral experiment show that most of the internal representations in Layer 1 were classified as low-level features (99%), such as contrast and edge features, whereas those in Layer 5 were classified as high-level features (55%), such as object parts and entire objects. Furthermore, the majority of the internal representations in the intermediate layers were classified as mid-level features (>57%) such as contour, shape, and texture features (Figure 3.3*C*,*D*). The receptive field complexities, invariances, and sizes of the convolutional voxel groups were significantly correlated with their layer assignments (Spearman's $\rho = 1$ and $p <$ 0.0167 for all properties, permutation test across convolutional

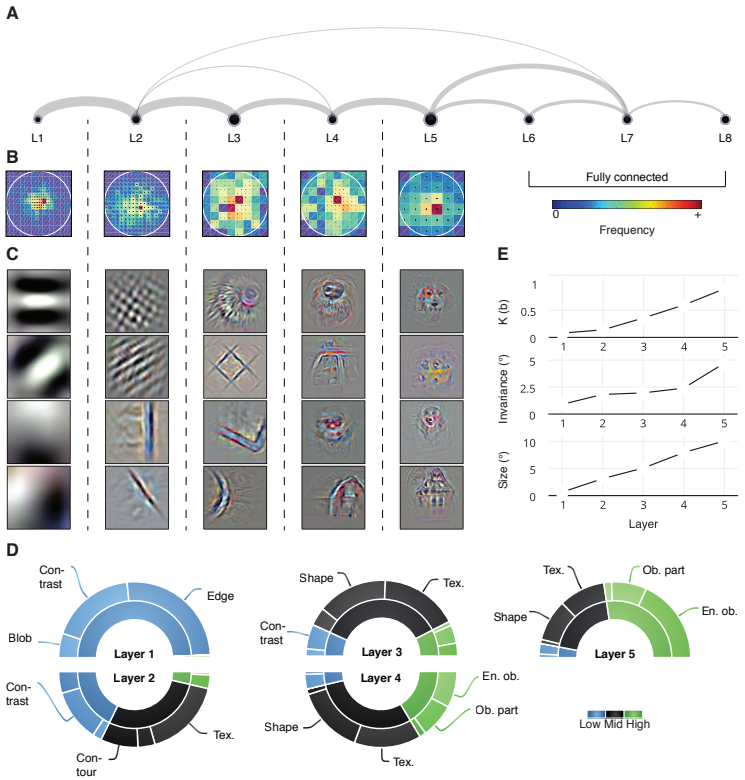



**Figure 3.3:** Properties of the voxel groups systematically change as a function of layer assignment. ***A***: Significant linear partial correlations between the predicted responses of each pair of voxel groups. Line widths are proportional to mean partial correlation coefficients across subjects. ***B***: Distribution of the receptive field centers for both subjects. ***C***: Example reconstructions of the internal representations of the convolutional layers. Reconstructions are enlarged, and automatic tone, contrast, and color enhancement are applied for visualization purposes. ***D***: Proportions of the internal representations of the convolutional layers that are assigned to low-level (blob, contrast, and edge), mid-level (contour, shape, and texture), and high-level (irregular pattern, object part, and entire object) feature classes. ***E***: Receptive field complexity (K), invariance, and size of the voxel groups.

layers; Figure 3.3*E*). Note that receptive field size is completely determined by the model's architecture.

## Voxel groups reveal a gradient in the complexity of neural representations

Different voxel groups were systematically clustered around different points on the cortical surface such that an increase in layer assignment was observed when moving from posterior to anterior points on the cortical surface (Figure 3.4*A*,*B*). We found a systematic overlap between these voxel groups and the visual areas on the main afferent pathway of the ventral stream. The mean layer assignment of the V1, V2, V4, and LO voxels was 1.8, 2.3, 3.0, and 5.0 for S1, and 1.6, 2.1, 3.9, and 5.2 for S2. The layer distributions of each pair of visual areas except V4 and LO of S2 were significantly different ($p < 6e\text{-}4$ for all pairs of visual areas except V4 and LO of S2; $p = 0.1206$ for V4 and LO of S2; Bonferroni correction for number of pairs, Mann–Whitney $U$ test across significant voxels within subjects). That is, most voxels assigned to shallow convolutional layers were located in early visual areas, whereas most voxels assigned to deep convolutional layers were located in downstream visual areas. Most voxels assigned to the fully connected layers were located in visual areas even more anterior to LO.

To characterize the distribution of the feature classes that best predict the voxels in each visual area, we assigned each significant voxel to one of the nine feature classes. That is, we repeated the encoding experiment by using each of the nine feature classes (rather than each of the eight layers) as input and assigning individual voxels to their optimal feature class according to maximal prediction accuracy computed using fivefold cross-validation on the training data (Figure 3.4*C*). It was found that V1 and LO were populated by voxels that were best predicted by low-level features ($p = 8e\text{-}80$, $\chi^2$ test across significant voxels and subjects) and high-level features ($p = 7e\text{-}19$, $\chi^2$ test across significant voxels and subjects), respectively. For example, the majority of V1 voxels (66%) were assigned to contrast and edge features, whereas the

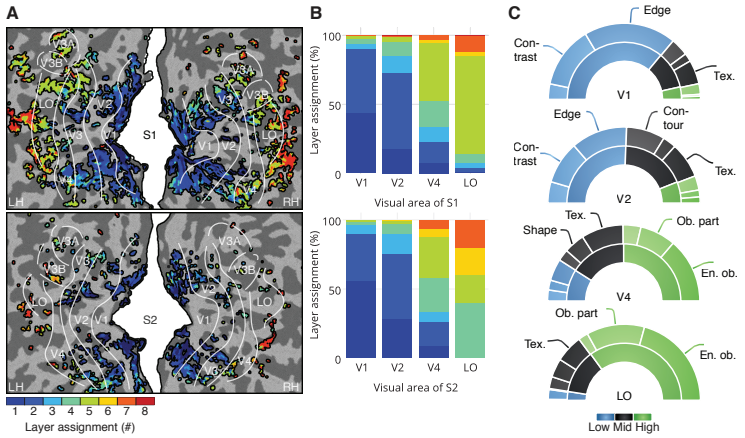



**Figure 3.4:** Layer assignments of the voxels systematically increase as a function of position on the occipital cortex. ***A***: Layer assignments of the significant voxels across occipital cortex ($p$ < 2e-6 for both subjects, Bonferroni corrected for number of voxels, Student's $t$ test across cross-validated training images within subjects). ***B***: Layer assignments of the significant voxels across V1, V2, V4, and LO ($p$ < 5e-8 for both subjects, Bonferroni corrected for number of layers and voxels, Student's $t$ test across cross-validated training images within subjects). ***C***: Proportions of voxels in areas V1, V2, V4, and LO that are assigned to low-level (blob, contrast, and edge), mid-level (contour, shape, and texture), and high-level (irregular pattern, object part, and entire object) feature classes.

majority of LO voxels were assigned to object parts and entire objects (66%). Compared with V1 voxels, a larger percentage of V2 voxels was best predicted by mid- and high-level features ($p$ = 8e-22, $\chi^2$ test across significant voxels and subjects). Similarly, a larger percentage of V4 than LO voxels was best predicted by low- and mid-level features ($p$ = 6e-7, $\chi^2$ test across significant voxels and subjects). For example, 32% of V2 voxels was assigned to contour and texture features, and 27% of V4 voxels was assigned to shape and texture features.

## Selectivity of voxels to individual feature maps reveals distributed representations

To investigate how individual features are represented across the cortical surface, we retrained a separate response model for each feature map/voxel combination. The selectivity of an individual voxel to a particular feature was defined as the cross-validated prediction accuracy of the corresponding response model on the training set. We found a many-to-many relationship between features and voxels (Figure 3.5*A*). That is, individual features accurately predicted multiple voxels and individual voxels were accurately predicted by multiple features. For features of either low or high complexity this relationship tended to be spatially confined to either upstream or downstream visual areas, respectively.

Next, we set out to understand whether individual visual areas revealed more fine-grained substructure. Biclustering of the prediction accuracy matrix revealed horizontal bands with fluctuating magnitude that point to features with similar information content, and vertical bands that point to clusters of voxels with congruent responses (Figure 3.5*B*). Constant magnitude vertical bands, for example, within areas V1 and V2, are likely caused by differences in SNR. In contrast, vertical bands with fluctuating magnitude, for example, within areas V4 and LO, point to clusters of voxels with unique response profiles that reflect functional specialization within individual visual areas.

## Comparison with control models

To further validate our model, we compared its prediction accuracies with those of different control models (Figure 3.6*A*). A comparison with the pretrained DNNs that made different architectural assumptions showed that there was no significant

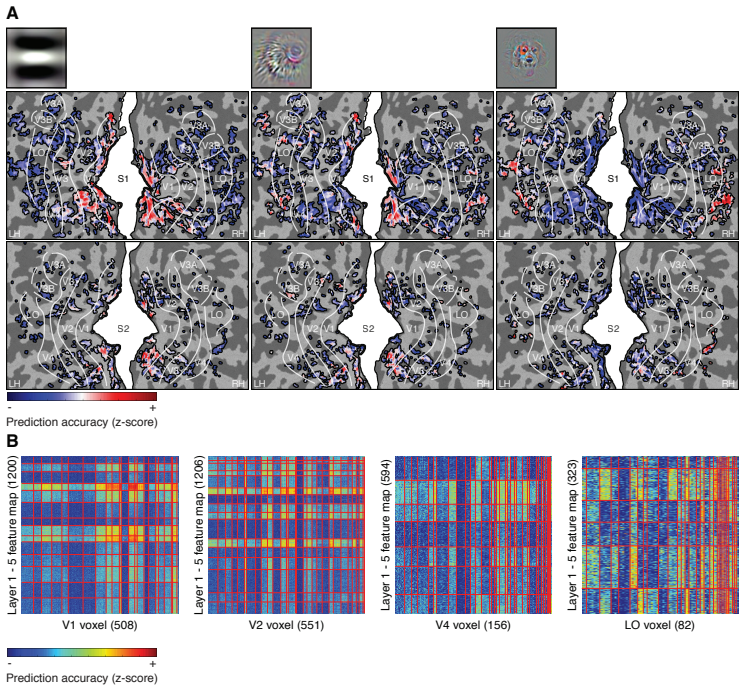



**Figure 3.5:** Voxels in different visual areas are differentially selective to feature maps in different layers. ***A***: Selectivity of the significant voxels in the occipital cortex to three distinct feature maps of varying complexity ($p < 2e\text{-}6$ for both subjects, Bonferroni corrected for number of voxels, Student's $t$ test across cross-validated training images within subjects). ***B***: Biclusters of hyperaligned voxels and feature maps. Horizontal and vertical red lines delineate the boundaries of clusters of feature maps and voxels, respectively. The rows and columns are thresholded such that each row and column contain at least one element that survives the threshold of $r^2 = 0.15$. The numbers in parentheses denote the number of remaining feature maps and voxels after thresholding.

difference between prediction accuracies of our model and the pretrained DNNs in any visual area ($p > 0.7267$ for all visual areas, two-sample $t$ test across significant voxels and subjects), and the pretrained DNNs maintained the representational gradient (Figure 3.6*B*). This demonstrates that our results are insensitive to exact architectural assumptions. However, the DNNs that had

the same architecture but randomly generated weights and biases were significantly outperformed by our model in each visual area ($p < 9e\text{-}18$ for all visual areas, two-sample $t$ test across significant voxels and subjects) and failed to maintain the representational gradient (Figure 3.6*B*). Furthermore, our model significantly outperformed the GWP model in each visual area ($p < 4e\text{-}14$ for all visual areas, two-sample $t$ test across significant voxels and subjects). These results demonstrate that optimizing for object categorization is an essential ingredient when explaining ventral stream responses.

## 3.4 Discussion

The present work used a DNN tuned for object categorization to probe neural responses to naturalistic stimuli. The results show that our approach accurately models these responses across the ventral stream. Moreover, by uncovering the internal representations of the DNN, we were able to quantify how different areas of the ventral stream respond to stimulus features of varying complexity.

### DNNs differentiate visual areas in terms of complexity, invariance, and receptive field size

By estimating the complexity of the internal representations of artificial neurons, we were able to quantitatively confirm the existence of a gradient in complexity of neural representations across visual areas on the main afferent pathway of the ventral stream. It was established that downstream areas code for increasingly complex stimulus features that belong to increasingly deep layers of the DNN. This representational gradient was further suppor-

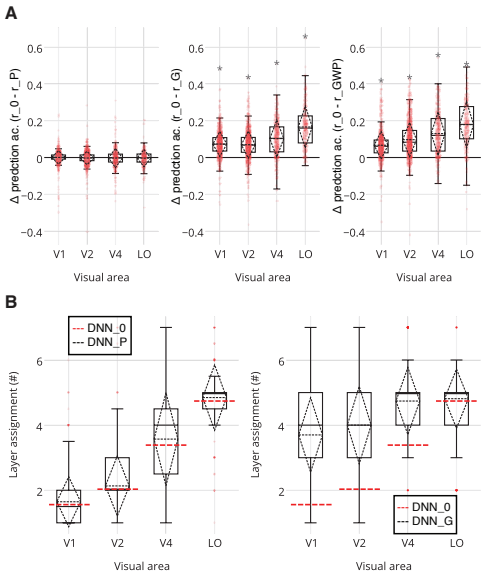



**Figure 3.6:** Our model performs similarly to the control models that are task optimized but outperforms those that are not task optimized across V1, V2, V4, and LO voxels of both subjects. ***A***: Comparison between the prediction accuracies for our model ($r_0$) with those for the pretrained DNN ($r_P$), random DNN ($r_R$), and GWP ($r_{GWP}$) models. Red dots denote the individual voxels. Asterisks indicate the visual areas where the prediction accuracies are significantly different. ***B***: Comparison between the layer assignments for our model ($DNN_0$) with those of the pretrained DNN ($DNN_P$) and random DNN ($DNN_R$) models. Red dots denote the individual voxels. Crosses indicate the mean layer assignments of the $DNN_0$ model.

ted by an increase in perceived feature complexity as tested by means of a behavioral experiment. These findings agree with the observation that semantic selectivity is organized as smooth gradients across cortex (Huth et al., 2012) and confirms earlier results on ventral stream responses to scrambled versus nonscrambled images (Grill-Spector et al., 1998). Our analyses further confirmed that downstream receptive fields become larger and more invariant (Smith, 2001; DiCarlo & Cox, 2007).

While most voxels respected the observed gradient in representational complexity, in a minority of voxels it was found that shallow DNN layers optimally code for downstream voxel responses and deep DNN layers code for upstream voxel responses (compare Figure 3.4). This is consistent with neurophysiological findings in primates that some downstream neurons are tuned to relatively simple features and some upstream neurons are tuned to relatively complex features (Desimone, Albright, Gross & Bruce, 1984; Hegdé & Essen, 2006). In general, our analyses reveal a many-to-many relationship between features and voxels. This implies that individual features are represented in a distributed manner across a patch of cortex and multiple features are superimposed on the same cortical expanse (Grill-Spector & Weiner, 2014). However, these observations might also be explained in part by confounding factors such as reliance on a limited amount of training data, indirect sampling of neural responses, and/or interactions between correlated stimulus features.

## High-throughput mapping and interpretation of neural representations

We view our work as an important step in the development of high-throughput analysis methods for mapping and interpretation of neural representations. We used complex, ecologically valid naturalistic stimuli (Felsen & Dan, 2005) to efficiently probe how thousands of individual stimulus features are represented across the cortical sheet. This can be contrasted with traditional approaches that typically make use of highly constrained artificial stimuli (Rust & Movshon, 2005). Mapping of individual stimulus features confirmed that low-level stimulus properties were mainly confined to early visual areas, whereas high-level stimulus properties were mostly represented in posterior inferior temporal areas. Furthermore, biclustering of feature-specific prediction accuracies revealed a more fine-grained functional specialization in down-

stream visual areas (Larsson & Heeger, 2006; Tanigawa, Lu & Roe, 2010).

The general applicability of DNN-based encoding models permits the investigation of neural representations in other visual areas (Agrawal et al., 2014) and in other brain regions involved in the representation of sensory information, such as the dorsal stream (Goodale & Milner, 1992) or multimodal association areas (Mesulam, 1998). Next to probing other brain regions, the framework lends itself to testing how representations change under various experimental manipulations. For example, it allows probing of pRF reconfigurations in the presence of top-down modulations such as changes in attention (Çukur et al., 2013) and task demand (Emadi & Esteky, 2014; McKee, Riesenhuber, Miller & Freedman, 2014), as a function of experience (Rainer, Lee & Logothetis, 2004; Çukur et al., 2013), or as a result of neurodegenerative disorders such as semantic dementia (Patterson, Nestor & Rogers, 2007). Finally, DNN-based decoding of stimuli from neural activity patterns may allow probing of internally generated percepts that occur during, e.g., imagery (Thirion et al., 2006), memory retrieval (Harrison & Tong, 2009), visual illusions (Kok & de Lange, 2014), and dreaming (Horikawa, Tamaki, Miyawaki & Kamitani, 2013), potentially offering novel insights into these more elusive cognitive processes.

## Accounting for unexplained variance

Even though DNNs yield state-of-the-art encoding performance, explained variance still remained low for a substantial number of voxels. This can be caused by several factors. First, our analyses revealed that low explained variance is caused in part by low SNR of observed voxel responses. That is, even though not all variance is explained, we are approaching the noise ceiling for particular voxels (Wu, David & Gallant, 2006). Second, stimulus features that drive particular voxels may only be present in a

minority of stimuli across the training set, precluding accurate response estimation. This is supported by the fact that prediction accuracy was positively correlated with the mean activity of neural network layers across the training set. Finally, prediction accuracy depends on the quality of the encoding model. Since the human brain obviously cannot be equated with a DNN that linearly maps stimulus features to observed BOLD responses, it is not surprising that residual variance remains. Hence, an important direction for future research is the development of more realistic encoding models.

One way to improve encoding performance is to develop feature models that outperform DNNs when it comes to capturing neural representations of low-, mid-, and high-level stimulus features. Arguably, unsupervised learning of statistical structure in our environment or the maximization of expected reward during reinforcement learning offer more biologically plausible explanations for the formation of receptive field properties. These alternative learning schemes might better account for the emergence of neural representations across cortex and may also be optimal for object categorization (Olshausen & Field, 1996; Schultz, Dayan & Montague, 1997). From a computational point of view it is not inconceivable that unsupervised or reinforcement learning schemes, which allow learning of multiple layers of increasingly complex stimulus features (Hinton, 2007; Mnih et al., 2015), will outperform DNN-based encoding models in explaining neural responses in particular brain regions.

Another avenue for further research is the development of more sophisticated response models. The current response model makes use of a linear mapping from a nonlinear feature representation onto peak BOLD amplitude. In reality, however, the mapping from stimulus features to responses should take into account the dynamics of vascular responses that result from changes in neuronal processing (Logothetis & Wandell, 2004; Norris, 2006). It is likely that encoding performance will further improve by using

more sophisticated (Pedregosa, Eickenberg, Ciuciu, Gramfort & Thirion, 2014) and/or biophysically realistic (Aquino, Robinson & Drysdale, 2014) response models.

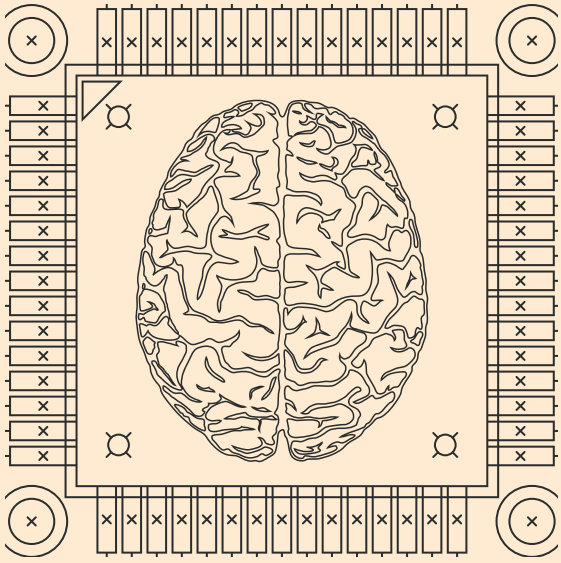
## Encoding models as hypotheses about brain function

While DNN-based encoding models are among the best computational models for explaining responses across the ventral stream, it does not follow that they provide a mechanistic account of perceptual processing in their biological counterparts. As one obvious example, our use of a strictly feedforward architecture cannot easily be reconciled with the feedback processing inherent to neural information processing (Hochstein & Ahissar, 2002). Rather, the utility of the encoding approach lies in testing whether a particular computational model outperforms alternative computational models when it comes to explaining observed data (Naselaris et al., 2011).

From a theoretical perspective, our DNN-based encoding model can be considered as implementing a hypothesis about the emergence of receptive field properties across the ventral stream (Fukushima, 1980). DNNs rely on the notion of object categorization to explain the emergence of a hierarchy of increasingly complex representations (Serre, Wolf, Bileschi, Riesenhuber & Poggio, 2007). The proposition that object categorization drives the formation of receptive field properties in the ventral stream is supported by the observation that performance-optimized hierarchical models can reliably predict single-neuron responses in area IT of the macaque monkey (Yamins et al., 2014). It is also substantiated by recent findings that DNNs better predict voxel responses in the human visual system and the representational geometry of IT responses in both macaques and humans, compared with other computational models (Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014). We extend these findings by showing that voxels in down-

stream areas of the ventral stream code for increasingly complex stimulus features that drive object categorization.

The goal of future computational models should be to improve on the present model, either by incorporating different assumptions or invoking other objective functions, reflecting alternative theories of brain function. Already at the earliest levels of visual processing, there remains ample room for debate as to what form an optimal computational model should take (Carandini, 2005). Notwithstanding the debate that remains, we subscribe to a model-based approach to cognitive neuroscience (Forstmann & Wagenmakers, 2015) in which theories about brain function are tested against each other by validating generative models on neural and/or behavioral data.



# 4

# Increasingly complex representations of natural movies across the dorsal stream are shared between subjects

This chapter is based on: Güçlü, U. and van Gerven, M. (2015). Increasingly complex representations of natural movies across the dorsal stream are shared between subjects. *NeuroImage*, 145(Part B):329–336. https://doi.org/10.1016/j.neuroimage.2015.12.036

## 4.1 Introduction

The human visual system is devoted to the analysis of increasingly complex properties of our environment as one moves from upstream to downstream visual areas. Traditionally, the ventral visual pathway is hypothesized to be devoted to object recognition and the dorsal visual pathway is thought to be devoted to motion processing and action recognition (Mishkin, Ungerleider & Macko, 1983; Haxby et al., 1991; Goodale & Milner, 1992).

An important question is what stimulus properties are processed as one traverses these pathways toward more downstream areas. Recently, we have shown that deep neural networks (DNNs) (Schmidhuber, 2015; LeCun et al., 2015) can be used to predict with high accuracy how voxels in different areas of the ventral stream respond to naturalistic stimuli (Güçlü & van Gerven, 2015). Moreover, this analysis revealed that artificial neurons in deeper hidden layers of the neural network gave better predictions for more downstream areas.

It remains unclear, however, whether DNNs can also be used to accurately predict neural responses across the dorsal stream up to and including area MT. Furthermore, if this property holds, an interesting secondary question is whether representations in particular visual areas are highly individualized or rather shared between subjects. If the latter is the case, then it may be possible to predict neural responses in a particular subject using computational models that are estimated using data from other subjects (Yamada, Miyawaki & Kamitani, 2015). Furthermore, if such a common representational space exists, decoding of stimuli from observed neural responses can be improved by combining data from multiple subjects.

The current paper addresses these questions using a sophisticated computational model, commonly referred to as an encod-

ing model (Naselaris et al., 2011). The encoding model, depicted in Figure 4.1, consists of a deep convolutional neural network (Fukushima, 1980) that nonlinearly maps stimuli to their constituent features, as well as a response model that linearly maps features to observed blood–oxygen-level-dependent (BOLD) responses.

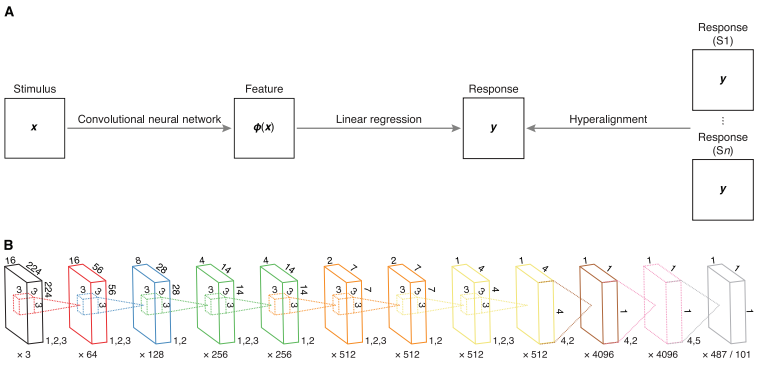



**Figure 4.1:** Framework that combines feature, response and representational space models. ***A***: Encoding model. ***B***: Convolutional neural network. Large boxes show a stimulus and feature maps, and numbers around them show their dimensionality. Similarly, small boxes and their projections show neurons, and numbers around them show their dimensionality. Number of feature maps and neurons in each layer is indicated below the boxes. Note that the dimensionality of a neuron in a fully-connected layer is the same as that of the feature maps in the previous layer. Each neuron filters the feature maps in the previous layer and returns the corresponding feature map in the current layer. Transformations in each layer are indicated bottom-right of the boxes: 1. Convolution. 2. Rectifier. 3. Max pooling. 4. Dot product. 5. Softmax function.

The deep neural network was trained using tens of thousands of action videos, yielding spatio-temporal filters that are important for action recognition, ostensibly yielding a representation suitable for probing dorsal stream responses. The linear mapping was estimated using data by Nishimoto et al. (2011b) in which subjects were watching natural movies. Estimation proceeded by first mapping data from different number of subjects to a com-

mon representational space and then averaging responses across subjects (Haxby et al., 2011). Next, deep neural network features were regressed onto averaged responses.

Using this framework, we were able to show (1) the existence of a correspondence between DNN layers and dorsal stream areas of individual subjects such that deeper layers better predict downstream areas and (2) the existence of a common representational space that can facilitate the estimation of common models for individual subject prediction such that responses of individual subjects to novel spatio-temporal stimuli can be predicted with models estimated from responses of other subjects in both encoding and decoding settings.

## 4.2 Material and methods

### Data set

We used the vim-2 data set (Nishimoto et al., 2014), which was originally published in (Nishimoto et al., 2011b). The experimental procedures are identical to those in (Nishimoto et al., 2011b). Briefly, the data set has twelve 600-s blocks of stimulus–response pairs in a training set and nine 60-s blocks of stimulus–response pairs in a test set. Stimuli are videos (128 px $\times$ 128 px or 20° $\times$ 20°, 15 FPS) that were drawn from various sources. Responses are BOLD responses (voxel size = 2 $\times$ 2 $\times$ 2.5 $mm^3$, TR = 1 s) that were acquired from occipital cortices of three subjects (S1, S2 and S3). Stimuli in the test set were repeated ten times. Responses in the test set were averaged across repetitions.

Stimuli in the data set were spatially downsampled to 112 px $\times$ 112 px and temporally upsampled to 16 FPS. Responses in the data set have already been preprocessed as described in (Nishimoto et al.,

2011b). Briefly, they have been realigned to compensate for motion, detrended to compensate for drift and z-scored. Additionally, the first six seconds of the blocks were discarded. No further preprocessing was performed.

Regions of interests were localized using the multifocal retinotopic mapping technique on retinotopic mapping data that were acquired in separate sessions (Hansen, David & Gallant, 2004). We restricted our analyses to dorsal stream visual areas (V1, V2, V3, V3A, V3B and MT).

## Hyperalignment

In addition to analyzing the data in the individual representational spaces, we analyzed them in a common representational space (Haxby et al., 2011). A representational space model that uses Procrustes transformation for hyperaligning the data of the individual subjects to the common representational space was estimated from the training set per cerebral hemisphere and visual area as follows: the common representational space was first set to the data of the individual subject that has the most number of voxels (Table 4.1). The common representational space was then iteratively updated. At each iteration, the data of the individual subjects were first projected to the common representational space. The common representational space was then set to the mean of the projections of the data of the individual subjects. After the final iteration, the data of the individual subjects were projected to the common representational space. *PyMVPA* (http://www.pymvpa.org) was used for representational space model estimation (Hanke et al., 2009).

# Encoding

### Feature model

We used a deep convolutional neural network for non-linearly transforming stimuli to multiple layers of hierarchical feature representations. The architecture of the DNN is identical to the C3D architecture in (Tran, Bourdev, Fergus, Torresani & Paluri, 2014). The architecture was developed for learning generic features for video analysis, building on previous insights in DNNs for image recognition. Here, we provide an overview of the architecture (for a more extensive treatment of DNNs, the reader is referred to Schmidhuber (2015), LeCun et al. (2015), Tran et al. (2014), Krizhevsky et al. (2012). The DNN has eight convolutional layers and three fully-connected layers of (artificial) neurons. There are 64, 128, 256, 256, 512, 512, 512, 512, 4096, 4096, and 487 (training) or 101 (fine-tuning) neurons in layers 1–11, respectively. Each neuron in the convolutional layers locally filters its input, non-linearly transforms it and returns a spatio-temporal map of feature responses (3D feature map). In contrast, each neuron in the fully-connected layers globally filters its input (dot product), non-linearly transforms it and returns a feature response (scalar or 1D feature map).

Let $\mathbf{x}$ be a second-long stimulus of size 112 px $\times$ 112 px $\times$ 16 frames $\times$ 3 color channels. Let $l$ and $m$ denote the index of a layer and a feature map, respectively. The filtered inputs (activations) of the $m$th feature map in the $l$th layer are given by:

$$\mathbf{a}_{l,m} = \begin{cases} \mathbf{x} * \mathbf{w}_{l,m} + \theta_{l,m} & \text{for } l = 1 \\ \mathbf{h}_{l-1}(\mathbf{a}_{l-1}) * \mathbf{w}_{l,m} + \theta_{l,m} & \text{for } 2 \leq l \leq 8 \\ \text{vec}(\ \mathbf{h}_{l\ -\ 1}(\ \mathbf{a}_{l\ -\ 1}\ )\ )^{\top}\mathbf{w}_{l,m}\ +\ \theta_{l,m} & \text{for } 9 \leq l \leq 11 \end{cases} \tag{4.1}$$

where $\mathbf{a}_l$ collects all activations in layer $l$, vec denotes vectorization, * denotes the convolution operator, $\mathbf{w}_{l,m}$ are the weights and $\theta_{l,m}$ is the bias term. The non-linear transformations $\mathbf{h}(\mathbf{z})$ are a composition of one or more of the following non-linear transformations:

- Rectification (layers 1–10):

$$h_i(\mathbf{z}) = \max(0, z_i) \tag{4.2}$$

- Max pooling (layers 1, 2, 4, 6, 8):

$$h_i(\mathbf{z}) = \begin{cases} \max\limits_{\substack{r \in [r_i - m, r_i + m] \\ c \in [c_i - m, c_i + m]}} z_{(r,c,f_i)} & \text{for } l = 1 \\ \max\limits_{\substack{r \in [r_i - m, r_i + m] \\ c \in [c_i - m, c_i + m] \\ f \in [f_i - m, f_i + m]}} z_{(r,c,f)} & \text{otherwise} \end{cases} \tag{4.3}$$

where $r_i$, $c_i$, and $f_i$ are the row, column and frame indices associated with element $z_i$ and where $\chi(r_i,c_i,f_i) = i$.

The final layer outputs are given by $\sigma(\mathbf{a}_l)$ where $\sigma_i(\mathbf{z}) = \exp z_i / \sum_j \exp z_j$ is the softmax function.

We used the pre-trained DNN in (Tran et al., 2014) to avoid training the DNN from scratch. The pre-trained DNN was trained on 1.1. $\times$ $10^6$ non-overlapping sports videos of size 128 px $\times$ 171 px $\times$ 2 s that were drawn from the Sports-1 M data set (Karpathy et al., 2014). The objective of training was to classify videos into one of

487 sports classes. Next, in order to explicitly optimize for action recognition, we fine-tuned the DNN on $8.4 \times 10^4$ non-overlapping action videos of size 128 px × 171 px × 1.1 s that were drawn from the UCF-101 data set (Soomro, Zamir & Shah, 2012). The objective of fine-tuning was to classify videos into one of 101 action classes. Fine-tuning did not change the architecture of the DNN except for the last layer where the number of neurons was changed from 487 to 101 (to account for the different number of classes). Stochastic gradient descent was used to train and fine-tune the DNN. The weight update rule is given by:

$$\mathbf{w}_0 \sim \mathcal{N}(0, 0.1\mathbf{I}) \tag{4.4}$$

$$\mathbf{w}_{i+1} = \mathbf{w}_i + \mathbf{v}_{i+1} \tag{4.5}$$

$$\mathbf{v}_{i+1} = \alpha \mathbf{v}^i - \beta\gamma\mathbf{w}_i - \gamma < \frac{\partial L}{\partial \mathbf{w}}, |\mathbf{w}_i >_{D_i} \tag{4.6}$$

where $\boldsymbol{w}_i$ are the weights, $\boldsymbol{v}_i$ are the weight updates, $D_i$ is the mini batch at the $i$th iteration, $\alpha$ is the momentum, $\beta$ is the weight decay, $\gamma$ is the learning rate, and $L$ is the objective function. Dropout with a probability of 0.5 was used to regularize layers 9 and 10 (Hinton et al., 2012). The bias update rule is analogous to the weight update rule. *Caffe* (http://caffe.berkeleyvision.org) was used for DNN training and fine-tuning (Jia et al., 2014).

For training, a momentum of $\alpha = 0.9$, a weight decay of $\beta = 5 \times 10^{-4}$, an initial learning rate of $\gamma = 3 \times 10^{-3}$ and a mini-batch size of $| D_i | = 30$ were used. The initial learning rate was changed by a factor of 0.5 after every $1.5 \times 10^5$ iterations. Training was stopped after $1.9 \times 10^6$ iterations. For fine-tuning, a momentum of $\alpha = 0.9$, a weight decay of $\beta = 5 \times 10^{-4}$, an initial learning rate of $\gamma = 3.3 \times 10^{-5}$ and a mini-batch size of $| D_i | = 10$ were used. The initial learning rate was changed by a factor of 0.1 after $3 \times 10^4$ iterations. Fine-tuning was stopped after $6 \times 10^4$

iterations. Videos in each training and fine-tuning mini-batch were randomly cropped to 112 px $\times$ 112 px $\times$ 16 frames and horizontally mirrored with a probability of 0.5. The hyperparameters in the training phase were the same as those in (Tran et al., 2014). The initial learning rate and the maximum iterations in the fine-tuning phase were selected on a validation set (a held-out part of the UCF-101 data set). The mini-batch size in the fine-tuning phase was less than that in the training phase because of GPU memory constraints.

We restricted our analyses to the neural network layers that use max pooling (layers 1, 2, 4, 6 and 8) and fully-connected layers (layers 9, 10 and 11). For simplicity, we refer to them as layers 1, 2, 3, 4, 5, 6, 7 and 8, respectively.

## Response model

Let $\mathbf{x}^t$ and $\mathbf{y}^t$ be a second-long stimulus–response pair between times $t$-1 and $t$ seconds. Furthermore, let $\mathbf{G}^t = \{\mathbf{g}_1(\mathbf{x}^t), \ldots, \mathbf{g}_8(\mathbf{x}^t)\}$, where we use $\mathbf{g}_l(\mathbf{x}^t)$ to denote the (vectorized) feature representations of an input $\mathbf{x}^t$ by the $l$th neural network layer. Recall that the sampling rate of $\mathbf{x}^t$ was 16 Hz whereas that of $\mathbf{y}^t$ was 1 Hz. We downsampled $\mathbf{G}^t$ to 1 Hz by keeping the first frame in each of the feature maps in all of the eight layers. Without loss of generality, we assume that $\mathbf{G}^t$ and $\mathbf{y}^t$ have zero mean and unit variance.

In encoding, our goal is to predict the most likely response given the feature representations:

$$\widehat{\mathbf{y}}^t = \arg\max_{\mathbf{y}} p(\mathbf{y}|L^{3-6}\mathbf{g}_1(\mathbf{x}^t), \ldots, L^{3-6}\mathbf{g}_8(\mathbf{x}^t))) \tag{4.7}$$

where $p$ is an encoding distribution, and $L^{3\text{-}6}$ is a lag operator that lags a time series by 3, 4, 5 and 6 s and concatenates these lagged time series to account for the hemodynamic delay.

For each voxel $k \in [1, K]$, we used regularized linear regression to define response models that were conditioned exclusively on the feature representations of individual layers $l$:

$$y_k^t = \beta_k^\top L^{3-6}\mathbf{g}_l(\mathbf{x}^t) + \varepsilon_k^t \tag{4.8}$$

where $\beta_k$ are the regression coefficients and $\varepsilon_v^t \sim \mathcal{N}(0, \sigma_k^2)$ is residual noise.

Let $\Phi_l = (L^{3\cdot 6}\mathbf{g}_l(\mathbf{x}^1), \ldots, L^{3\cdot 6}\mathbf{g}_l(\mathbf{x}^N))^\top$ and $\boldsymbol{Y} = (\mathbf{y}^1, \ldots, \mathbf{y}^N)^\top$ be a set of $N$ second-long feature transformed stimulus–response pairs. We analytically minimized the $L^2$-penalized least squares loss function to estimate the regression coefficients:

$$\widehat{\mathbf{B}} = (\boldsymbol{\Phi}_l^\top \boldsymbol{\Phi}_l + \lambda \mathbf{I})^{-1} \boldsymbol{\Phi}_l^\top \mathbf{Y} \tag{4.9}$$

where $\widehat{\mathbf{B}} = [\widehat{\beta}_1, \ldots, \widehat{\beta}_K]$ and $\lambda \geq 0$ is a regularization parameter.

We used nested leave-one-block-out cross-validation on the test set for response model selection, estimation, and evaluation. Response model selection amounts to selecting a layer $l \in [1,\ldots,8]$ and a regularization parameter $\lambda$ from a grid that is defined as in (Güçlü & van Gerven, 2014) per voxel. The outer cross-validation had nine folds, and the inner cross-validation had eight folds.

## Decoding

Let $\mathcal{X}$ be a set consisting of two second-long stimuli that contains a target stimulus $\mathbf{x}^t$. Let $r(\mathbf{u},\mathbf{v})$ denote the correlation between vectors $\mathbf{u}$ and $\mathbf{v}$. In decoding, our goal is to identify $\mathbf{x}^t$ from $\mathcal{X}$ given $\boldsymbol{y}^{t+4}$ where the four second lag accounts for the hemodynamic delay (Nishimoto et al., 2011b):

$$\widehat{\mathbf{x}}^t = \arg\max_{\mathbf{x}^t \in \mathcal{X}} r(\mathbf{y}^{t+4}, \widehat{\mathbf{y}}^{t+4}) \tag{4.10}$$

where $\widehat{\mathbf{y}}^{t+4} = \arg\max_{\mathbf{y}^{t+4}} p(\mathbf{y}^{t+4}|\mathbf{g}_1(\mathbf{x}^t), \ldots, \mathbf{g}_8(\mathbf{x}^t))$ was predicted as described in the encoding section except for the use of a different stimulus–response mapping:

$$y_k^{t+4} = \beta_k^\top \mathbf{g}_l(\mathbf{x}^t) + \varepsilon_k^t \tag{4.11}$$

that was employed to identify stimuli on a per-second basis.

To identify videos longer than one second, we $z$-transformed the $r$-values of each one second of the videos, averaged the $z$-values and inverse $z$-transformed the $z$-values. Similar to encoding, we used nested leave-one-block-out cross-validation on the test set for model selection, estimation, and evaluation.

## Analyses

Both for encoding and decoding we considered several different analyses, as depicted in Figure 4.2.

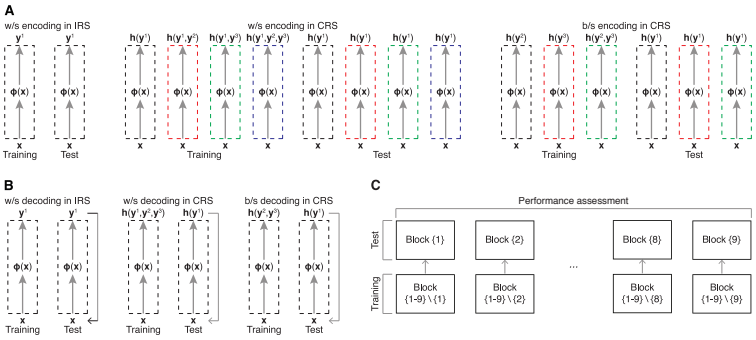



**Figure 4.2:** Illustration of the different encoding (***A***) and decoding (***B***) analyses, and cross-validation (***C***) performed for one subject (S1). Analyses were repeated for the remaining two subjects (S2 and S3). $\mathbf{y}^i$ denotes the data of S$i$, $\mathbf{h}(\mathbf{y}^1, \ldots, \mathbf{y}^M)$ denotes the (mean) hyperaligned data of subjects $S1,\ldots,SM$ and \ denotes set difference. Notation w/s and b/s stands for within- and between-subject analyses, respectively. IRS and CRS stand for individual and common representational space analyses, respectively.

For encoding/decoding in individual representational space, the model that was trained on the data of a subject was tested on the data of the same subject. For example, the model trained on the data of 'S1' was tested on the data of 'S1' but not 'S2' or 'S3'.

For analyses in common representational space, separate models were trained on (i) the hyperaligned data of 'S1', 'S2' and 'S3' (three one-subject models), (ii) mean hyperaligned data of 'S1 and S2', 'S1 and S3' and 'S2 and S3' (three two-subject models), and (iii) mean hyperaligned data of 'S1, S2 and S3' (one three-subject model).

For encoding analyses, cases (i), (ii) and (iii) were considered for within-subject analyses and cases (i) and (ii) were considered for between-subject analyses. For decoding analyses, only case (iii) was considered for within-subject analyses and only case (ii) was considered for between-subject analyses. For within-subject analyses, the models were separately tested on the hyperaligned

data of the individual subjects that were used to train them. For example, the model trained on the hyperaligned data of 'S1' was tested on the hyperaligned data of 'S1' but not 'S2' or 'S3'. For between-subject analyses, the models were separately tested on the hyperaligned data of the individual subjects that were *not* used to train them. For example, the model trained on the hyperaligned data of 'S1' was tested on the hyperaligned data of 'S2' and 'S3' but not 'S1'.

Note that we used the training set exclusively for hyperalignment and the test set exclusively for encoding and decoding. Also note that we used nested leave-one-block-out cross-validation on the test set for model selection, estimation, and evaluation irrespective of the exact analysis (encoding/decoding, individual/common representational space, within -/between subject, one -/two -/three subjects).

## Performance assessment

We quantified the encoding performance of a voxel in terms of the prediction accuracy, which we define as Pearson's correlation between the observed and predicted responses to 486 second-long videos. We quantified the encoding performance of a visual area as the median encoding performance of the voxels in the visual area. The prediction accuracies of the voxels were corrected for their noise ceilings (Kay et al., 2013a) and negative values were set to zero.

Decoding performance was quantified as the area under the receiver operating characteristic (ROC) curve (AUC). Concretely, $r$-values between each pair of observed and predicted responses were computed. This resulted in an $N \times N$ correlation matrix where $N$ is the number of videos. The on-diagonal elements were taken as the positive instances, and the off-diagonal elements

were taken as the negative instances. Finally, an ROC curve was constructed and AUC was taken as the decoding performance.

Intuitively, AUC for identifying a target stimulus from a set that contains the target stimulus and an arbitrary stimulus given the observed response to the target stimulus can be interpreted as the probability that the correlation between the observed and predicted responses to the target stimulus is higher than that between the observed response to the target stimulus and the predicted response to the arbitrary stimulus.

For optimal decoding performance, a subset of voxels is typically selected (Güçlü & van Gerven, 2015; Nishimoto et al., 2011b; Güçlü & van Gerven, 2014; Kay et al., 2008) since voxels for which the encoding model has low predictive power degrades decoding performance. This becomes especially important when multiple models are being compared since suboptimal number of voxels can lead to misleading performance differences between the models. Therefore, we selected 500 (baseline), 1500 (within-subject) and 1000 (between-subject) voxels that had the highest noise ceilings since these resulted in the highest decoding performance among those that were tested (Figure 4.6).

Permutation tests were used for comparing an (encoding or decoding) model against chance level. First, data were randomly permuted over time for 200 times. Then, a separate model was trained and tested for each of the 200 permutations. Finally, the $p$-value was taken to be the fraction of the 200 permutations whose encoding/decoding performance was greater than the actual encoding/decoding performance. The performance was considered significant if the $p$-value was less than 0.05 (Bonferroni corrected for number of visual areas) for encoding and 0.05 (Bonferroni corrected for number of video durations) for decoding.

# 4.3 Results

## Within-subject encoding in individual representational space

In the first experiment, we analyzed the data of the individual subjects in their own representational spaces (Figure 4.3).

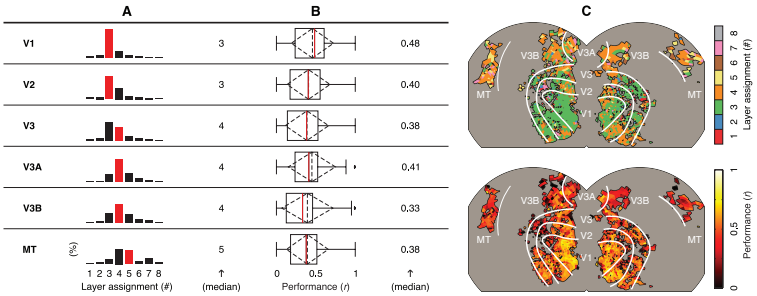



**Figure 4.3:** Results of within-subject encoding in individual-subject voxel space. Representations of increasingly downstream dorsal stream voxels of individual subjects can be modeled as linear combinations of representations that are learned by increasingly deep layers of DNNs. Results are pooled across subjects. ***A***: Layer assignments of voxels. Red bars show medians. ***B***: Prediction accuracies of voxels. Red lines show medians. Boxes show lower and upper quartiles. Whiskers show data closest to the 1.5 interquartile range of lower and upper quartiles. Dots show data outside 1.5 interquartile range of lower and upper quartiles. Dashed lines show means and standard deviations. ***C***: Projections of layer assignments and prediction accuracies of voxels to cortical flat maps of S1. *FreeSurfer* (http://surfer.nmr.mgh.harvard.edu) and *MrTools* (http://gru.stanford.edu/doku.php/mrtools/overview) were used for cortical surface reconstruction and flattening, respectively.

We first estimated layer assignments of the voxels. The layer assignment of a voxel was defined as the layer of the DNN that was most predictive of its responses. The distributions of the layer assignments of the voxels in all of the pairs of visual areas except

V3A and V3B were significantly different ($p < 0.001$, $p > 0.5$, Mann–Whitney $U$ test). There was a systematic change in the layer assignments of the voxels moving from V1 to MT. The proportions of layer 3 voxels decreased, and those of layer 5 voxels monotonically increased. In contrast, the proportions of layer 4 voxels monotonically increased moving from V1 to V3A and monotonically decreased moving from V3A to MT. The median layer assignment of the V1, V2 and V3 voxels was 3, that of V3A and V3B voxels was 4, and that of MT voxels was 5. More than 81% of the voxels were assigned to these layers.

We then estimated prediction accuracies of the voxels. The median prediction accuracies of the voxels in all of the visual areas were significantly above chance level ($p < 0.001$, permutation test, Bonferroni correction). The median overall prediction accuracy of the voxels was 0.41. The V1 voxels had the highest median prediction accuracy (0.48). The V3B voxels had the lowest median prediction accuracy (0.33). Furthermore, there was a significant correlation between the prediction accuracies of all voxels and their noise ceilings ($r = 0.3183$, Pearson's $r$, $p < 0.001$, Student's $t$-test) as well as the mean prediction accuracies of the voxels assigned to a layer and the mean activity of the layer ($r = 0.6616$, Pearson's $r$, $p < 0.001$, Student's $t$-test), which provides a partial explanation as to why responses of voxels in some visual areas (e.g. V1) can be better predicted than those in other visual areas (e.g. MT).

These results suggest that representations of increasingly downstream dorsal stream voxels of individual subjects can be modeled as linear combinations of representations that are learned by increasingly deep layers of DNNs optimized for action recognition.

## Within- and between-subject encoding in common representational space

In the second experiment, we analyzed the data of the individual subjects in the common representational space (Figure 4.4). Models were trained on hyperaligned data of individual subjects, mean hyperaligned data of two subjects or those of three subjects. In the within-subject encoding case, they were tested on hyperaligned data of individual subjects that were used to train them. In the between-subject encoding case, they were tested on hyperaligned data of individual subjects that were *not* used to train them.

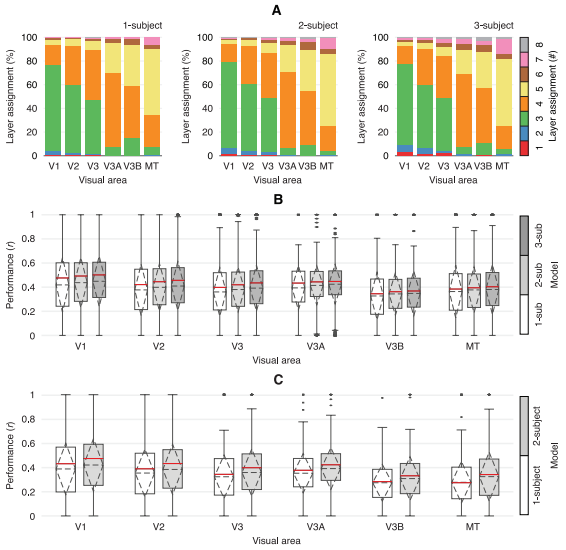



**Figure 4.4:** Results of within- and between-subject encoding in common representational space. A common representational space underlies dorsal stream responses across multiple subjects. Results are pooled across subjects. ***A***: Layer assignments of hyperaligned voxels. ***B***: Within-subject encoding performance. ***C***: Between-subject encoding performance. Box plots have the same representation as those in Figure 4.1*B*.

The trends in the layer assignments and prediction accuracies that were found in the individual representational spaces were found

also in the common representational space. The median layer assignment of the hyperaligned V1, V2 and V3 voxels was 3, that of the hyperaligned V3A and V3B voxels was 4, and that of the hyperaligned MT voxels was 5. The median layer assignments were not significantly different ($p > 0.5$, Mann–Whitney $U$ test). In the within-subject encoding case, the median overall prediction accuracies of the hyperaligned voxels were 0.42, 0.44 and 0.45 for models trained on data from one, two or three subjects, respectively. In the between-subject encoding case, the median overall prediction accuracies of the hyperaligned voxels were 0.37, 0.42 for models trained on data from one and two subjects, respectively. In both cases, prediction accuracies significantly increased as the hyperaligned data of more subjects were used to train the models ($p < 0.001$, binomial test).

These results suggest that a common representational space underlies dorsal stream responses across multiple subjects. Furthermore, they show that a common encoding model that combines feature, response and representational space models can generalize to responses of multiple or even unseen individual subjects to novel stimuli.

## Within- and between-subject decoding in common representational space

In the third experiment, we tested the extent to which a common decoder that combines feature, response and representational space models can decode responses of individual subjects to novel stimuli (Figure 4.5). Specifically, we analyzed the following two cases in addition to the baseline decoding case (no hyperalignment): (i) a within-subject decoding case in which the common decoder was trained on the mean hyperaligned data of 'S1, S2 and S3', and tested on the hyperaligned data of 'S1', 'S2' and 'S3', respectively. (ii) A between-subject decoding case in which the common decoder was trained on the mean hyperaligned data

of ‘S1 and S2’, ‘S1 and S3’ and ‘S2 and S3’, and tested on the hyperaligned data of ‘S3’, ‘S2’ and ‘S1’, respectively.

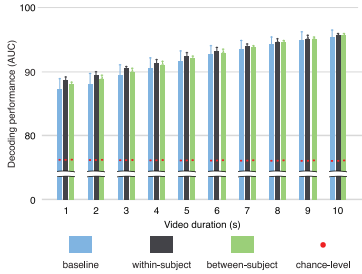



**Figure 4.5:** Results of decoding in individual representational space (baseline) as well as within- and between-subject decoding in common representational space. A common decoder that combines feature, response and representational space models can decode responses of multiple or unseen individual subjects to novel stimuli. Results are pooled across subjects. Bars show AUCs of separately identifying each of the $\sim$ 486 $N$-second-long target videos from a set that contains the target video and one of the remaining $\sim$ 485 $N$-second-long candidate videos ($N$ = 1,...,10) given the observed response to the target video. The exact number of videos depended on $N$ (# videos = 9 $\times$ (55-$N$)). Error bars show $\pm$ 1 SEM.

We estimated AUCs of separately identifying each of the $\sim$ 486 $N$-second-long target videos from a set that contains the target video and one of the remaining $\sim$ 485 $N$-second-long candidate videos ($N$ = 1,...,10) given the observed response to the target video. The exact number of videos depended on $N$ (# videos = 9 $\times$ (55-$N$)). Note that slightly different response models were used in this experiment than those that were used in the previous experiments.

All of the AUCs were significantly above the chance level of $\sim$ 50% ($p < 0.001$, permutation test, Bonferroni correction) and correlated with the video durations ($\rho = 1$, Spearman’s rho, $p < 0.001$, Student’s $t$-test). One-second-long videos were identified with an AUC of 87.22% (baseline), 88.58% (within-subject) and 87.98% (between-subject). Ten-second-long videos were identified with an

AUC of 95.70% (baseline), 95.89% (within-subject) and 95.80% (between-subject). While the within-subject AUCs were consistently higher than the between-subject AUCs, and the between-subject AUCs were consistently higher than the baseline AUCs, they were not significantly different ($p > 0.05$, $Z$-test (Hanley & McNeil, 1982)).

### Control analyses

In the final experiment we evaluated how hyperalignment data size affects encoding and decoding performance (Figure 4.7. We compared the results obtained in the main experiments (hyperalignment with the entire training set — 12 blocks of data) with those obtained after hyperalignment with part of the training set — one, five and eight blocks of data. We found a slight but significant decrease in encoding performance when only one or five blocks of data were used for hyperalignment ($p < 0.05$, binomial test). For the other cases, there was no significant decrease in performance as the size of the data used for hyperalignment was reduced ($p > 0.05$, binomial test for encoding, $Z$-test for decoding).

## 4.4 Discussion

Our results have shown that DNNs optimized for action recognition can be used to accurately predict how dorsal stream areas respond to dynamically changing naturalistic stimuli. Furthermore, as for ventral stream areas (Güçlü & van Gerven, 2015), we have shown that layer depth of the best coding layer and the location of an area across the dorsal stream are positively correlated. This indicates that more downstream areas code for increasingly complex features of their environment.

The work presented here adds to the body of evidence that deep convolutional neural networks yield state-of-the-art predictions of visual cortical responses. Whereas previous work focused on understanding visual cortical responses with DNNs optimized for object recognition (Güçlü & van Gerven, 2015; Khaligh-Razavi & Kriegeskorte, 2014; Cadieu et al., 2014; Yamins et al., 2014; Eickenberg, 2015; Agrawal et al., 2014), the current results show that dorsal stream responses are accurately captured by DNNs optimized for action recognition. That is, we generalize the previous results on models that learn spatial representations to models that learn spatio-temporal representations, which more accurately captures the properties of motion selective receptive fields (Rust & Movshon, 2005; Nishimoto et al., 2011a). However, the proposed framework still lacks the ability to model long-term dependencies. Recent developments in recurrent neural networks in general and long short-term memory architectures in particular (Hochreiter & Schmidhuber, 1997; Greff, Srivastava, Koutník, Steunebrink & Schmidhuber, 2015) can be used to tackle the problem of long-term dependencies of neural representations.

While we found a correspondence between DNN layers and dorsal stream visual areas, it remains a challenge to visualize and/or parameterize these representations. Our approach can be complemented with recent developments in visualizing DNN representations, for example with deconvolution or backpropagation (Zeiler & Fergus, 2013; Simonyan, Vedaldi & Zisserman, 2013; Mahendran & Vedaldi, 2014; Springenberg, Dosovitskiy, Brox & Riedmiller, 2014; Dosovitskiy & Brox, 2015), to get a better grip of what these representations look like. Furthermore, these developments can also be utilized for reconstructing perceived stimuli instead of identifying them from a set of candidate stimuli.

One assumption made in the present study is that the voxels can be modeled as linear combinations of features in the same layer. However, it could be the case that representations that span multiple layers might better explain certain voxel responses. The

approach of fitting a single regularized linear regression model per voxel using all DNN layers can be used to tackle this question. However, this currently remains computationally infeasible due to the high dimensionality of the employed feature representations. Alternatively, features can be clustered either automatically or manually based on criteria other than the layers they belong to in order to identify representations that span multiple layers (Güçlü & van Gerven, 2015).

One interesting question that remains to be tackled is whether the representations of other sensory areas can be modeled with DNNs. This question can be addressed using our approach by training DNNs on other sensory stimuli and testing them on other sensory areas. For example, DNNs that are trained on optic flow (Simonyan & Zisserman, 2014) and audio (Dieleman & Schrauwen, 2014) were shown to learn representations that are useful for different categorization tasks. Integrating such DNNs in encoding models can provide insights into how other sensory stimuli are represented across subjects.

Additionally, it was shown that a common representational space can facilitate the estimation of common models for individual subject prediction. By averaging responses to the same stimulus across subjects in the common representational space, we found that prediction accuracy increases while retaining the same findings regarding the existence of a representational gradient (cf. Figs. 4.4*A* and *B*). Furthermore, due to the correspondence between different subjects, data from one subject could be used to predict regional responses in other subjects at high accuracy, particularly for more upstream areas (cf. Figure 4.4*C*). Finally, it was shown that a common model trained on data in the common representational space could be used to identify which movie fragment was seen by multiple or even unseen subjects at high accuracy (cf. Figure 4.5).

The fact that encoding and decoding works just as well in the common representational space across subjects has important implications. First, it supports the hypothesis that specific brain regions subserve the same function across subjects (Haxby et al., 2011). Second, it boosts within-subject performance. Third, it allows transfer learning where responses from previously unseen subjects can be predicted from other subjects' data in both encoding and decoding settings.

Estimation of accurate encoding models typically requires many hours of data per subject. Subjects usually find it difficult to stay in the scanner for many runs or long sessions, degrading data quality. Therefore, another advantage of encoding models that include a representational space model is that this problem can be partially overcome by either transfer learning (Yamada et al., 2015), or estimating and/or evaluating encoding models from less data per subject by pooling data of multiple subjects.

It should also be noted that data of three subjects were analyzed in this study. Our focus was to investigate the existence of (i) a correspondence between DNN layers and dorsal stream areas of individual subjects and (ii) a common representational space that can facilitate the estimation of common models for individual subject prediction. Therefore, all analyses and tests were performed separately for individual subjects. Our results revealed the existence of such a correspondence and a common representational space within these subjects. However, it does not follow that our conclusions can necessarily be generalized to the whole population, which would require a larger group study.

Concluding, our findings show that deep neural networks optimized for action recognition are able to predict responses across the dorsal stream, providing further support for the notion that areas along the dorsal pathway are optimized for action recognition. Future research should provide insights into the question

whether optimization for action recognition is necessary or rather just sufficient for modeling dorsal stream responses.

## Supplementary material

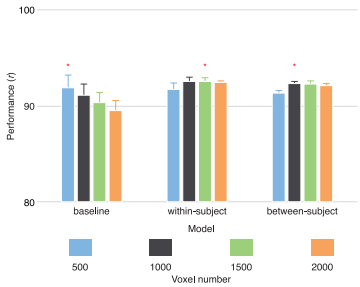



**Figure 4.6:** Influence of number of used voxels on decoding performance. Results are pooled across subjects. Red asterisks show maximums. Error bars show ± 1 SEM.

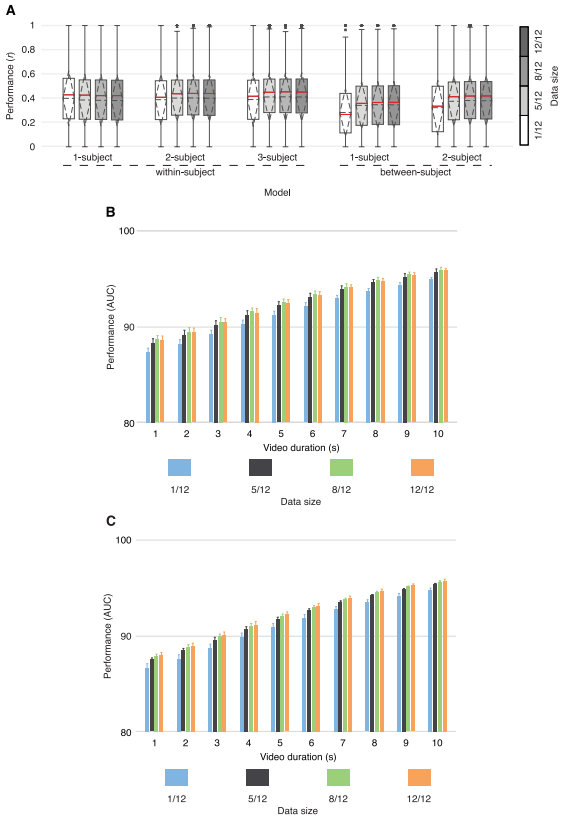
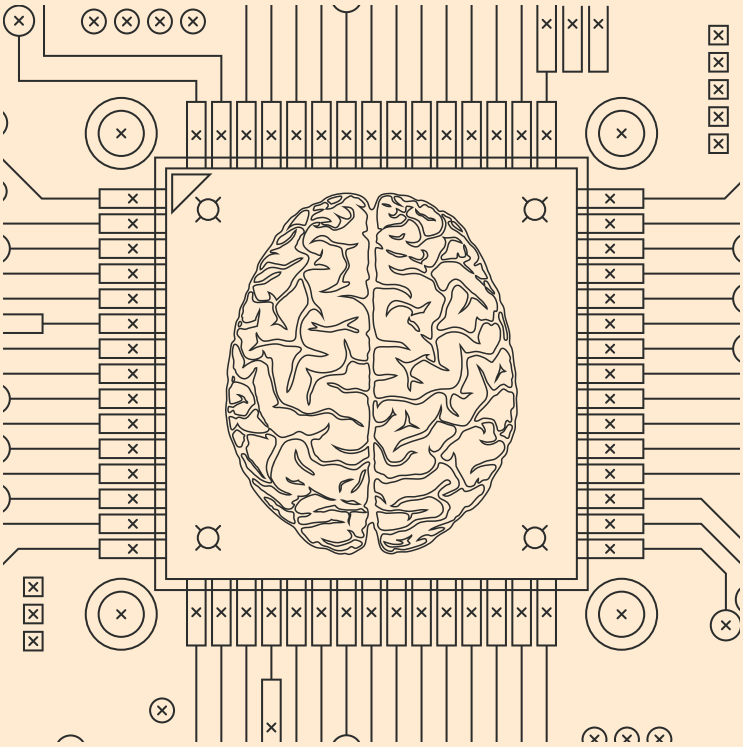



**Figure 4.7:** Effect of hyperalignment data size on encoding and decoding performance. Results are pooled across subjects. ***A***: Within- and between-subject encoding performance. Box plots have the same representation as those in Figure 4.3*B*. ***B***: Within-subject decoding performance. Error bars show ± 1 SEM. ***C***: Between-subject decoding performance. Error bars show ± 1 SEM.

**Table 4.1:** Number of voxels per cerebral hemisphere and visual area. Bold numbers show the dimensionality of the corresponding cerebral hemisphere and visual area in the common representational space.

| | **Left hemisphere** | | | | | | **Right hemisphere** | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **V1** | **V2** | **V3** | **V3A** | **V3B** | **MT** | **V1** | **V2** | **V3** | **V3A** | **V3B** | **MT** |
| S1 | 494 | 726 | 598 | 92 | **104** | **197** | 514 | 781 | 562 | 160 | **152** | **152** |
| S2 | 470 | 733 | **734** | 135 | 83 | 83 | 573 | **928** | **670** | **202** | 140 | 116 |
| S3 | **653** | **746** | 504 | **164** | 88 | 166 | **713** | 650 | 637 | 117 | 138 | 64 |



# 5 Brains on beats

This chapter is based on: Güçlü, U., Thielen, J., Hanke, M., and van Gerven, M. (2016). Brains on beats. In *Neural Information Processing Systems*. https://papers.nips.cc/paper/6222-brains-on-beats.pdf

## 5.1 Introduction

The human sensory system is devoted to the processing of sensory information to drive our perception of the environment (Schwartz & Krantz, 2015). Sensory cortices are thought to encode a hierarchy of ever more invariant representations of the environment (Fuster, 2003). A research question that is at the core of sensory neuroscience is what sensory information is processed as one traverses the sensory pathways from the primary sensory areas to higher sensory areas.

The majority of the work on auditory cortical representations has remained limited to understanding the neural representation of hand-designed low-level stimulus features such as spectro-temporal models (Santoro et al., 2014), spectro-location models (Moerel, Martino, Uğurbil, Yacoub & Formisano, 2015), timbre, rhythm, tonality (Alluri et al., 2012; Alluri et al., 2013; Toiviainen, Alluri, Brattico, Wallentin & Vuust, 2014) and pitch (Patterson, Uppenkamp, Johnsrude & Griffiths, 2002) or high-level representations such as music genre (Casey, Thompson, Kang, Raizada & Wheatley, 2012) and sound categories (Staeren, Renvall, Martino, Goebel & Formisano, 2009). For example, Santoro et al. (2014) found that a joint frequency-specific modulation transfer function predicted observed fMRI activity best compared to frequency-nonspecific and independent models. They showed specificity to fine spectral modulations along Heschl's gyrus (HG) and anterior superior temporal gyrus (STG), whereas coarse spectral modulations were mostly located posterior-laterally to HG, on the planum temporale (PT), and STG. Preference for slow temporal modulations was found along HG and STG, whereas fast temporal modulations were observed on PT, and posterior and medially adjacent to HG. Also, it has been shown that activity in STG, somatosensory cortex, the default mode network, and cerebellum are sensitive to timbre, while amygdala, hippocampus and insula are more sensitive to rhythmic and tonality features (Alluri et al.,

2012; Toiviainen et al., 2014). However these efforts have not yet provided a complete algorithmic account of sensory processing in the auditory system.

Since their resurgence, deep neural networks (DNNs) coupled with functional magnetic resonance imaging (fMRI) have provided a powerful approach to form and test alternative hypotheses about what sensory information is processed in different brain regions. On one hand, a task-optimized DNN model learns a hierarchy of nonlinear transformations in a supervised manner with the objective of solving a particular task. On the other hand, fMRI measures local changes in blood-oxygen-level dependent hemodynamic responses to sensory stimulation. Subsequently, any subset of the DNN representations that emerge from this hierarchy of nonlinear transformations can be used to probe neural representations by comparing DNN and fMRI responses to the same sensory stimuli. Considering that the sensory systems are biological neural networks that routinely perform the same tasks as their artificial counterparts, it is not inconceivable that DNN representations are suitable for probing neural representations.

Indeed, this approach has been shown to be extremely successful in visual neuroscience. To date, several task-optimized DNN models were used to accurately model visual areas on the dorsal and ventral streams (Yamins et al., 2014; Agrawal et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Cadieu et al., 2014; Horikawa & Kamitani, 2015; Cichy et al., 2016; Seibert et al., 2016; Cichy, Khosla, Pantazis & Oliva, 2017), revealing representational gradients where deeper neural network layers map to more downstream areas along the visual pathways (Güçlü & van Gerven, 2015; Güçlü & van Gerven, 2017). Recently, Kell et al. (2016) has shown that deep neural networks trained to map speech excerpts to word labels could be used to predict brain responses to natural sounds. Here, deeper neural network layers were shown to map to auditory brain regions that were more distant from primary auditory cortex.

In the present work we expand on this line of research where our aim was to model how the human brain responds to music. We achieve this by probing neural representations of music features across the superior temporal gyrus using a deep neural network optimized for music tag prediction. We used the representations that emerged after training a DNN to predict tags of musical excerpts as candidate representations for different areas of STG in representational similarity analysis. We show that different DNN layers correspond to different locations along STG such that anterior STG is shown to be more sensitive to low-level stimulus features encoded in shallow DNN layers whereas posterior STG is shown to be more sensitive to high-level stimulus features encoded in deep DNN layers.

## 5.2 Materials and methods

### *MagnaTagATune* dataset

We used the MagnaTagATune dataset (Law, West, Mandel, Bay & Downie, 2009) for DNN estimation. The dataset contains 25.863 music clips. Each clip is a 29 seconds long excerpt from 5223 songs from 445 albums from 230 artists. Each excerpt is supplied with a vector of binary annotations of 188 tags. These annotations are obtained by humans playing the two-player online TagATune game. In this game, the two players are either presented with the same or a different audio clip. Subsequently, they are asked to come up with tags for their specific audio clip. Afterward, players view each other's tags and are asked to decide whether they were presented the same audio clip. Tags are only assigned when more than two players agreed. The annotations include tags like 'singer', 'no singer', 'violin', 'drums', 'classical', 'jazz', et cetera. We restricted our analysis on this dataset to the top 50 most popular tags to ensure that there is enough training data for

each tag. Parts 1-12 were used for training, part 13 was used for validation and parts 14-16 were used for testing.

## *Studyforrest* dataset

We used the existing studyforrest dataset (Hanke et al., 2015) for representational similarity analysis. The dataset contains fMRI data on the perception of musical genres. Twenty participants (age 21-38 years, mean age 26.6 years), with normal hearing and no known history of neurological disorders, listened to twenty-five 6 second, 44.1 kHz music clips. The stimulus set comprised five clips per each of the five following genres: Ambient, Roots Country, Heavy Metal, 50s Rock 'n Roll, and Symphonic. Stimuli were selected according to the procedure of Casey et al. (2012). The Ambient and Symphonic genres can be considered as non-vocal and the others as vocal. Participants completed eight runs, each with all twenty-five clips.

Ultra-high-field (7 Tesla) fMRI images were collected using a Siemens MAGNETOM scanner, T2*-weighted echo-planar images (gradient-echo, repetition time (TR) = 2000 ms, echo time (TE) = 22 ms, 0.78 ms echo spacing, 1488 Hz/Px bandwidth, generalized auto-calibrating partially parallel acquisition (GRAPPA), acceleration factor 3, 24 Hz/Px bandwidth in phase encoding direction), and a 32 channel brain receiver coil. Thirty-six axial slices were acquired (thickness = 1.4 mm, 1.4 $\times$ 1.4 mm in-plane resolution, 224 mm field-of-view (FOV) centered on the approximate location of Heschl's gyrus, anterior-to-posterior phase encoding direction, 10% inter-slice gap). Along with the functional data, cardiac and respiratory traces, and a structural MRI were collected. In our analyses, we only used the data from the 12 subjects (Subjects 1, 3, 4, 6, 7, 9, 12, 14–18) with no known data anomalies as reported in (Hanke et al., 2015).

The anatomical and functional scans were preprocessed as follows: Functional scans were realigned to the first scan of the first run and next to the mean scan. Anatomical scans were coregistered to the mean functional scan. Realigned functional scans were slice-time corrected to correct for the differences in image acquisition times between the slices. Realigned and slice-time corrected functional scans were normalized to MNI space. Finally, a general linear model was used to remove noise regressors derived from voxels unrelated to the experimental paradigm and estimate BOLD response amplitudes (Kay, Rokem, Winawer, Dougherty & Wandell, 2013c). We restricted our analyses to the superior temporal gyrus (STG).

## Deep neural networks

We developed three task-optimized DNN models for tag prediction. Two of the models comprised five convolutional layers followed by three fully-connected layers (DNN-T model and DNN-F model). The inputs to the models were 96000-dimensional time (DNN-T model) and frequency (DNN-F model) domain representations of six second-long audio signals, respectively. One of the models comprised two streams of five convolutional layers followed by three fully connected layers (DNN-TF model). The inputs to the streams were given by the time and frequency representations. The outputs of the convolutional streams were merged and fed into first fully-connected layer. Figure 5.1 illustrates the architecture of the one-stream models.

We used Adam (Kingma & Ba, 2014) with parameters $\alpha = 0.0002$, $\beta_1 = 0.5$, $\beta_2 = 0.999$, $\epsilon = 1e^{-8}$ and a mini batch size of 36 to train the models by minimizing the binary cross-entropy loss function. Initial model parameters were drawn from a uniform distribution as described in (Glorot & Bengio, 2010). Songs in each training mini-batch were randomly cropped to six seconds (96000 samples). The epoch in which the validation performance

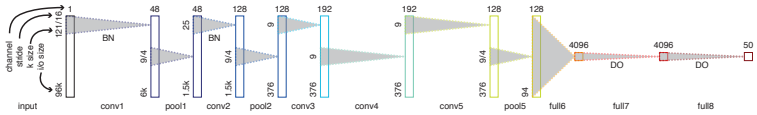



**Figure 5.1:** Architecture of the one-stream models. First seven layers are followed by parametric softplus units (McFarland, Cui & Butts, 2013), and the last layer is followed by sigmoid units. The architecture is similar to that of AlexNet (Krizhevsky, Sutskever & Hinton, 2012) except for the following modifications: (i) The number of convolutional kernels are halved. (ii) The (convolutional and pooling) kernels and strides are flattened. That is, an $n \times n$ kernel is changed to an $n^2 \times 1$ kernel and an $m \times m$ stride is changed to an $m^2 \times 1$ stride. (iii) Local response normalization is replaced with batch normalization (Ioffe & Szegedy, 2015). (iv) Rectified linear units are replaced with parametric softplus units with initial $\alpha = 0.2$ and initial $\beta = 0.5$. (v) Softmax units are replaced with sigmoid units.

was the highest was taken as the final model (53, 12 and 12 for T, F and TF models, respectively). The DNN models were implemented in Keras (Chollet, 2015).

Once trained, we first tested the tag prediction performance of the models and identified the model with the highest performance. To predict the tags of a 29 second long song excerpt in the test split of the MagnaTagaTune dataset, we first predicted the tags of 24 six-second-long overlapping segments separated by a second and averaged the predictions.

We then used the model with the highest performance for non-linearly transforming the stimuli to eight layers of hierarchical representations for subsequent analyses. Note that the artificial neurons in the convolutional layers locally filtered their inputs (1D convolution), nonlinearly transformed them and returned temporal representations per stimulus. These representations were further processed by averaging them over time. In contrast, the artificial neurons in the fully-connected layers globally filtered their inputs (dot product), non-linearly transformed them and re-

turned scalar representations per stimulus. These representations were not further processed. These transformations resulted in $n$ matrices of size $m \times p_i$ where $n$ is the number of layers (8), $m$ is the number of stimuli (25) and $p_i$ is the number of artificial neurons in the $i$th layer (48 or 96, 128 or 256, 192 or 384, 192 or 384, 128 or 256, 4096, 4096 and 50 for $i = 1, \ldots, 8$, respectively).

## Representational similarity analysis

We used Representational Similarity Analysis (RSA) (Kriegeskorte, 2008) to investigate how well the representational structures of DNN model layers match with that of the response patterns in STG. In RSA, models and brain regions are characterized by $n \times n$ representational dissimilarity matrices (RDMs), whose elements represent the dissimilarity between the neural or model representations of a pair of stimuli. In turn, computing the overlap between the model and neural RDMs provides evidence about how well a particular model explains the response patterns in a particular brain region. Specifically, we performed a region of interest analysis as well as a searchlight analysis by first constructing the RDMs of STG (target RDM) and the model layers (candidate RDM). In the ROI analysis, this resulted in one target RDM per subject and eight candidate RDMs. For each subject, we correlated the upper triangular parts of the target RDM with the candidate RDMs (Spearman correlation). We quantified the similarity of STG representations with the model representations as the mean correlation. For the searchlight analysis, this resulted in 27277 target RDMs (each derived from a spherical neighborhood of 100 voxels) and 8 candidate RDMs. For each subject and target RDM, we correlated the upper triangular parts of the target RDM with the candidate RDMs (Spearman correlation). Then, the layers which resulted in the highest correlation were assigned to the voxels at the center of the corresponding neighborhoods. Finally, the layer assignments were averaged over the subjects and the result was taken as the final layer assignment of the voxels.

## Control models

To evaluate the importance of task optimization for modeling STG representations, we compared the representational similarities of the entire STG region and the task-optimized DNN-TF model layers with the representational similarities of the entire STG region and two sets of control models.

The first set of control models transformed the stimuli to the following 48-dimensional model representations[1]:

- Mel-frequency spectrum (mfs) representing a mel-scaled short-term power spectrum inspired by human auditory perception where frequencies organized by equidistant pitch locations. These representations were computed by applying (i) a short-time Fourier transform and (ii) a mel-scaled frequency-domain filterbank.

- Mel-frequency cepstral coefficients (mfccs) representing both broad-spectrum information (timbre) and fine-scale spectral structure (pitch). These representations were computed by (i) mapping the mfs to a decibel amplitude scale and (ii) multiplying them by the discrete cosine transform matrix.

- Low-quefrency mel-frequency spectrum (lq_mfs) representing timbre. These representations were computed by (i) zeroing the high-quefrency mfccs, (ii) multiplying them by the inverse of discrete cosine transform matrix and (iii) mapping them back from the decibel amplitude scale.

- High-quefrency mel-frequency spectrum (hq_mfs) representing pitch. These representations were computed by (i) zeroing the low-quefrency mfccs, (ii) multiplying them by

[1] These are provided as part of the *studyforrest* dataset (Hanke et al., 2015).

the inverse of discrete cosine transform matrix and (iii) mapping them back from the decibel amplitude scale.

The second set of control models were 10 random DNN models with the same architecture as the DNN-TF model, but with parameters drawn from a zero mean and unit variance multivariate Gaussian distribution.

## 5.3 Results

In the first set of experiments, we analyzed the task-optimized DNN models. The tag prediction performance of the models for the individual tags was defined as the area under the receiver operator characteristics (ROC) curve (AUC).

We first compared the mean performance of the models over all tags (Figure 5.2). The performance of all models was significantly above chance level ($p \ll 0.001$, Student's $t$-test, Bonferroni correction). The highest performance was achieved by the DNN-TF model (0.8939), followed by the DNN-F model (0.8905) and the DNN-T model (0.8852). To the best of our knowledge, this is the highest tag prediction performance of an *end-to-end* model evaluated on the same split of the same dataset (Dieleman & Schrauwen, 2014). The performance was further improved by averaging the predictions of the DNN-T and DNN-F models (0.8982) as well as those of the DNN-T, DNN-F and DNN-TF models (0.9007). To the best of our knowledge, this is the highest tag prediction performance of *any* model (ensemble) evaluated on the same split of the same dataset (Dieleman & Schrauwen, 2013, 2014; van den Oord, Dieleman & Schrauwen, 2014). For the remainder of the analyses, we considered only the DNN-TF model since it achieved the highest single-model performance.

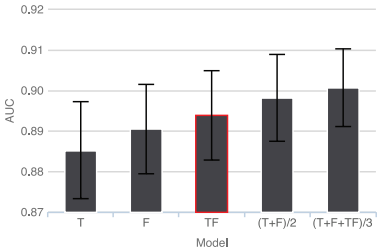



**Figure 5.2:** Tag prediction performance of the task-optimized DNN models. Bars show AUCs over all tags for the corresponding task-optimized DNN models. Error bars show ± SE. All pairwise differences are significant except for the pairs 1 and 2, and 2 and 3 ($p < 0.05$, paired-sample $t$-test, Bonferroni correction).

We then compared the performance of the DNN-TF model for the individual tags (Figure 5.3). Visual inspection did not reveal a prominent pattern in the performance distribution over tags. The performance was not significantly correlated with tag popularity ($p > 0.05$, Student's $t$-test). The only exception was that the performance for the positive tags were significantly higher than that for the negative tags ($p \ll 0.001$, Student's $t$-test).

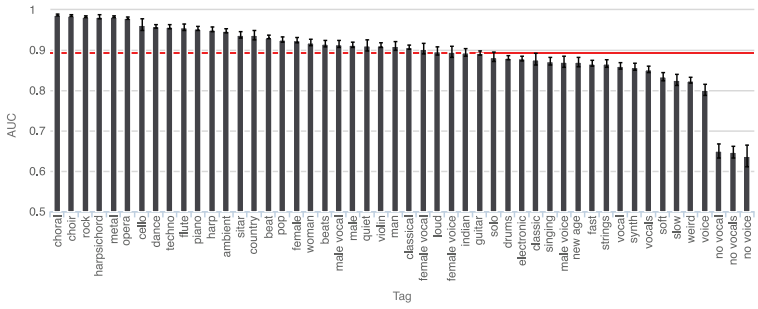



**Figure 5.3:** Tag prediction performance of the task-optimized DNN-TF model. Bars show AUCs for the corresponding tags. Red band shows the mean ± SE for the task-optimized DNN-TF model over all tags.

In the second set of experiments, we analyzed how closely the representational geometry of STG is related to the representational geometries of the task-optimized DNN-TF model layers.

First, we constructed the candidate RDMs of the layers (Figure 5.4). Visual inspection revealed similarity structure patterns that became increasingly prominent with increasing layer depth. The most prominent pattern was the non-vocal and vocal subdivision.

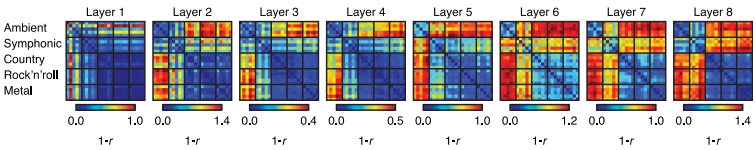



**Figure 5.4:** RDMs of the task-optimized DNN-TF model layers. Matrix elements show the dissimilarity (1 - Spearman's $r$) between the model layer representations of the corresponding trials. Matrix rows and columns are sorted according to the genres of the corresponding trials.

Second, we performed a region of interest analysis by comparing the reference RDM of the entire STG region with the candidate RDMs (Figure 5.5). While none of the correlations between the reference RDM and the candidate RDMs reached the noise ceiling (expected correlation between the reference RDM and the RDM of the true model given the noise in the analyzed data (Kriegeskorte, 2008)), they were all significantly above chance level ($p < 0.05$, signed-rank test with subject RFX, FDR correction). The highest correlation was found for Layer 1 (0.6811), whereas the lowest correlation was found for Layer 8 (0.4429).

Third, we performed a searchlight analysis (Kriegeskorte, Goebel & Bandettini, 2006) by comparing the reference RDMs of multiple STG voxel neighborhoods with the candidate RDMs (Figure 5.6). Each neighborhood center was assigned a layer such that the corresponding target and candidate RDM were maximally correlated. This analysis revealed a systematic change in the mean layer assignments over subjects along STG. They increased from anterior STG to posterior STG such that most voxels in the region of the transverse temporal gyrus were assigned to the shallower layers and most voxels in the region of the angular gyrus were

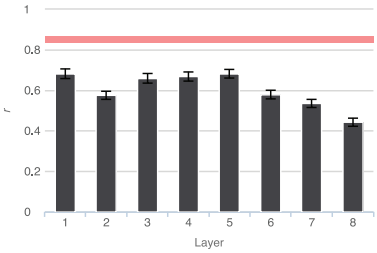



**Figure 5.5:** Representational similarities of the entire STG region and the task-optimized DNN-TF model layers. Bars show the mean similarity (Spearman's $r$) of the target RDM and the corresponding candidate RDMs over all subjects. Error bars show $\pm$ SE. Red band shows the expected representational similarity of the STG and the true model given the noise in the analyzed data (noise ceiling). All pairwise differences are significant except for the pairs 1 and 5, 2 and 6, and 3 and 4 ($p < 0.05$, signed-rank test with subject RFX, FDR correction).

assigned to the deeper layers. The corresponding mean correlations between the target and the candidate RDMs decreased from anterior to posterior STG.

In order to quantify the gradient in layer assignment, we correlated the mean layer assignment of the STG voxels in each coronal slice with the slice position, which was taken to be the slice number. As a result, it was found that layer and position are significantly correlated for the voxels along the anterior - posterior STG direction ($r = 0.7255$, Pearson's $r$, $p \ll 0.001$, Student's $t$-test). Furthermore, the mean correlations between the target and the candidate RDMs for the majority (85.53%) of the STG voxels were significant ($p < 0.05$, signed-rank test with subject RFX, FDR correction for the number of voxels followed by Bonferroni correction for the number of layers). However, the correlations of many voxels at the posterior end of STG were not highly significant in contrast to their central counterparts and ceased to be significant as the (multiple comparisons corrected) critical value was decreased from 0.05 to 0.01, which reduced the number of voxels surviving the critical value from 85.53% to 75.32%. Nevertheless,

the gradient in layer assignment was maintained even when the voxels that did not survive the new critical value were ignored ($r = 0.7332$, Pearson's $r$, $p \ll 0.001$, Student's $t$-test).

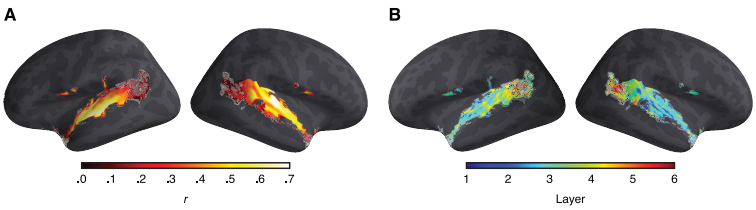



**Figure 5.6:** Representational similarities of the spherical STG voxel clusters and the task-optimized DNN-TF model layers. Only the STG voxels that survived the (multiple comparisons corrected) critial value of 0.05 are shown. Those that did not survive the critical value of 0.01 are indicated with transparent white masks and black outlines. ***A***: Mean representational similarities over subjects. ***B***: Mean layer assignments over subjects.

These results show that increasingly posterior STG voxels can be modeled with increasingly deeper DNN layers optimized for music tag prediction. This observation is in line with the visual neuroscience literature where it was shown that increasingly deeper layers of DNNs optimized for visual object and action recognition can be used to model increasingly downstream ventral and dorsal stream voxels (Güçlü & van Gerven, 2015; Güçlü & van Gerven, 2017). It also agrees with previous work showing a gradient in auditory cortex with DNNs optimized for speech-to-word mapping (Kell et al., 2016). It would be of particular interest to compare the respective gradients and use the music and speech DNNs as each other's control model such as to disentangle speech- and music-specific representations in auditory cortex.

In the last set of experiments, we analyzed the control models. We first constructed the RDMs of the control models (Figure 5.7). Visual inspection revealed considerable differences between the RDMs of the task-optimized DNN-TF model and those of the control models.

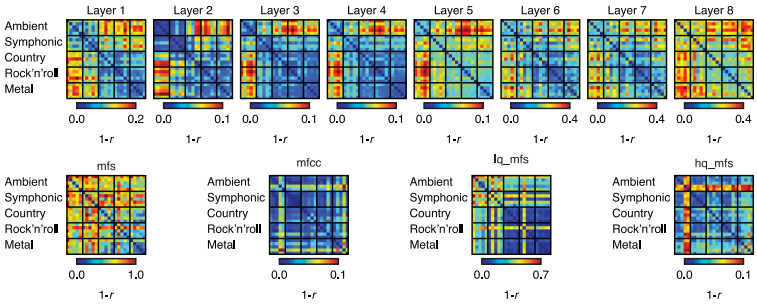



**Figure 5.7:** RDMs of the random DNN model layers (top row) and the baseline models (bottom row). Matrix elements show the dissimilarity (1 - Spearman's $r$) between the model layer representations of the corresponding trials. Matrix rows and columns are sorted according to the genres of the corresponding trials.

We then compared the similarities of the task-optimized candidate RDMs and the target RDM versus the similarities of the control RDMs and the target RDM (Figure 5.8). The layers of the task-optimized DNN model significantly outperformed the corresponding layers of the random DNN model ($\Delta r = 0.21$, $p < 0.05$, signed-rank test with subject RFX, FDR correction) and the four baseline models ($\Delta r = 0.42$ for mfs, $\Delta r = 0.21$ for mfcc, $\Delta r = 0.44$ for lq_mfs and $\Delta r = 0.34$ for hq_mfs, signed-rank test with subject RFX, FDR correction). Furthermore, we performed the searchlight analysis with the random DNN model to determine whether the gradient in layer assignment is a consequence of model architecture or model representation. We found that the random DNN model failed to maintain the gradient in layer assignment ($r = -0.2175$, Pearson's $r$, $p = 0.0771$, Student's $t$-test), suggesting that the gradient is in the representation that emerges from task optimization.

These results show the importance of task optimization for modeling STG representations. This observation also is line with visual neuroscience literature where similar analyses showed the

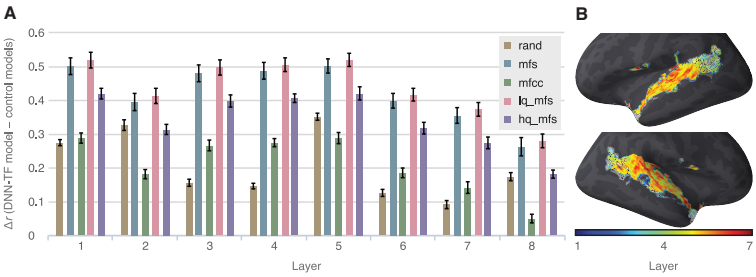
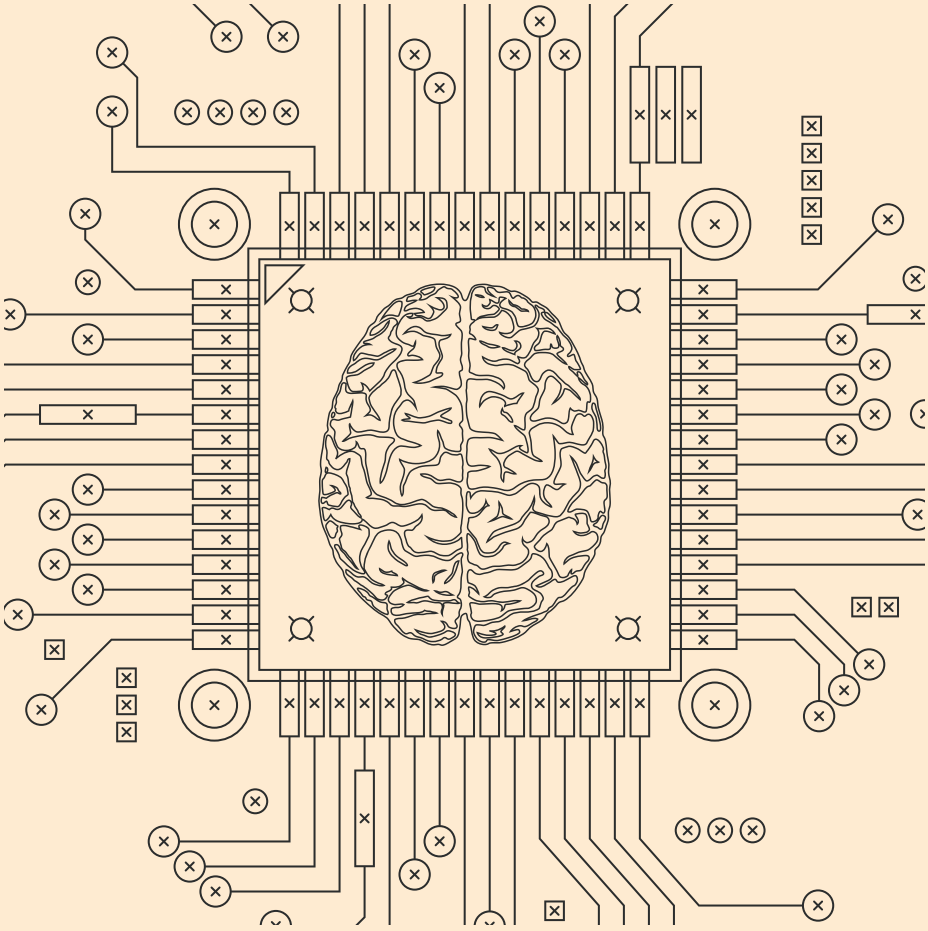



**Figure 5.8:** Control analyses. ***A***: Representational similarities of the entire STG region and the task-optimized DNN-TF model versus the representational similarities of the entire STG region and the control models. Different colors show different control models: Random DNN model, mfs model, mfcc model, lq_mfs model and hq_mfs model. Bars show mean similarity differences over subjects. Error bars show ± SE. ***B***: Mean layer assignments over subjects for the random DNN model. Voxels, masks and outlines are the same as those in Figure 5.6.

importance of task optimization for modeling ventral stream representations (Güçlü & van Gerven, 2015; Seibert et al., 2016).

## 5.4 Conclusion

We showed that task-optimized DNNs that use time and/or frequency domain representations of music achieved state-of-the-art performance in various evaluation scenarios for automatic music tagging. Comparison of DNN and STG representations revealed a representational gradient in STG with anterior STG being more sensitive to low-level stimulus features (shallow DNN layers) and posterior STG being more sensitive to high-level stimulus features (deep DNN layers). These results, in conjunction with previous results on the visual and auditory cortical representations, suggest the existence of multiple representational gradients that process increasingly complex conceptual information as we traverse sensory pathways of the human brain.



# 6

# Modeling the dynamics of human brain activity with recurrent neural networks

This chapter is based on: Güçlü, U. and van Gerven, M. (2017). Modeling the dynamics of human brain activity with recurrent neural networks. *Frontiers* in Computational Neuroscience, 11:7. https://doi.org/10.3389/fncom.2017.00007

## 6.1 Introduction

Encoding models (Naselaris et al., 2011) are used for predicting brain activity in response to naturalistic stimuli (Felsen & Dan, 2005) with the objective of understanding how sensory information is represented in the brain. Encoding models typically comprise two main components. The first component is a feature model that nonlinearly transforms stimuli to features (i.e., the independent variables used in fMRI time series analyses). The second component is a response model that linearly transforms features to responses. While encoding models have been successfully used to characterize the relationship between stimuli in different modalities and responses in different brain regions, their performance usually falls short of the expected performance of the true encoding model given the noise in the analyzed data (noise ceiling). This means that there usually is unexplained variance in the analyzed data that can be explained solely by improving the encoding models.

One way to reach the noise ceiling is the development of better feature models. Recently, there has been extensive work in this direction. One example is the use of convolutional neural network representations of natural images or natural movies to explain low-, mid- and high-level representations in different brain regions along the ventral (Agrawal et al., 2014; Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Yamins et al., 2014; Güçlü & van Gerven, 2015; Cichy et al., 2016) and dorsal streams (Güçlü & van Gerven, 2017; Eickenberg et al., 2017) of the human visual system. Another example is the use of manually constructed or statistically estimated representations of words and phrases to explain the semantic representations in different brain regions (Mitchell et al., 2008; Huth et al., 2012; Murphy et al., 2012; Fyshe et al., 2013; Güçlü & van Gerven, 2015; Nishida et al., 2015).

Another way to reach the noise ceiling is the development of better response models. There is a long history of estimating hemodynamic response functions (HRFs) in fMRI time series modeling. The standard general linear (convolution) model used in procedures like statistical parametric mapping (SPM) expands the HRF in terms of orthogonal kernels or temporal basis functions that have been motivated in terms of Volterra expansions. Indeed, commonly used software packages such as the SPM software have (hidden) facilities to model second-order Volterra kernels that enable modeling of non-linear hemodynamic effects such as saturation. In reality, the transformation from stimulus features to observed responses is exceedingly complex because of various temporal dependencies that are caused by neurovascular coupling (Logothetis & Wandell, 2004; Norris, 2006) and other more elusive cognitive or neural factors.

Here, our objective is to develop a model that can be trained end to end, captures temporal dependencies and processes arbitrary input sequences for time-continuous fMRI experiments such as watching movies, listening to music or playing video games. Such time-continuous designs are characterized by the absence of discrete experimental events as those found in their block or event-related counterparts. To this end, we use recurrent neural networks (RNNs) as response models in the encoding framework. Recently, RNNs in general and two RNN variants—long short-term memory (Hochreiter & Schmidhuber, 1997) and gated recurrent units (Cho et al., 2014)—in particular have been shown to be extremely successful in various tasks that involve processing of arbitrary input sequences such as handwriting recognition (Graves et al., 2009; Graves, 2013), language modeling (Sutskever, Martens & Hinton, 2011; Graves, 2013), machine translation (Cho et al., 2014) and speech recognition (Sak, Senior & Beaufays, 2014). These models use their internal memories to capture the temporal dependencies that are informative about solving the task at hand. That is, these models base their predictions not only to the information available at a given time, but also to the information that

was available in the past. They accomplish this by maintaining an explicit or implicit representation of the past input sequences and use it to make their predictions at each time point. If these models can be used as response models in the encoding framework, it will open a new window into modeling brain activity in response to sensory stimuli since the brain activity is modulated by long temporal dependencies.

While the use of RNNs in the encoding framework has been proposed a number of times (Güçlü & van Gerven, 2015; Güçlü & van Gerven, 2017; Kriegeskorte, 2015; Yamins & DiCarlo, 2016b, 2016b), these proposals mainly focused on using RNNs as feature models. In contrast, we have framed our approach in terms of response models used in characterizing distributed or multivariate responses to stimuli in the encoding framework. The key thing that we bring to the table is a generic and potentially useful response model that transforms features to observed (hemodynamic) responses. From the perspective of conventional analyses of functional magnetic resonance imaging (fMRI) time series, this response model corresponds to the convolution model used to map stimulus features (e.g., the presence of biological motion) to fMRI responses. In other words, the stimulus features correspond to conventional stimulus functions that enter standard convolution models of fMRI time series (e.g., the GLM used in statistical parametric mapping).

In brief, we know that the transformation from neuronal responses to fMRI signals is mediated by neuronal and hemodynamic factors that can always be expressed in terms of a non-linear convolution. A general form for these convolutions has been previously considered in the form of Volterra kernels or functional Taylor expansions (Friston, Mechelli, Turner & Price, 2000). Crucially, it is also well known that RNNs are universal non-linear approximators that can reproduce any Volterra expansion (Wray & Green, 1994). This means that we can use RNNs as an inclusive and flexible way to parameterize the convolution of stimulus features

generating hemodynamic responses. Furthermore, we can use RNNs to model not just response of a single voxel but distributed responses over multiple voxels. Having established the parametric form of this convolution, the statistical evidence or significance of each regionally specific convolution can then be assessed using standard (cross-validation) machine learning techniques by comparing the accuracy of the convolution when applied to test data after optimization with training data.

We test our approach by comparing how well a family of RNN models and a family of ridge regression models can predict blood-oxygen-level dependent (BOLD) hemodynamic responses to high-level and low-level features of natural movies using cross-validation. We show that the proposed recurrent neural network models can significantly outperform the standard ridge regression models and accurately estimate hemodynamic response functions by capturing temporal dependencies in the data.

## 6.2 Material and methods

### Data set

We analyzed the vim-2 data set (Nishimoto et al., 2014), which was originally published by Nishimoto et al. (2011a). The experimental procedures are identical to those in (Nishimoto et al., 2011a). Briefly, the data set has twelve 600 s blocks of stimulus and response sequences in a training set and nine 60 s blocks of stimulus and response sequences in a test set. The stimulus sequences are videos (512 px $\times$ 512 px or 20° $\times$ 20°, 15 FPS) that were drawn from various sources. The response sequences are BOLD responses (voxel size = 2 $\times$ 2 $\times$ 2.5 $mm^3$, TR = 1 s) that were acquired from the occipital cortices of three subjects (S1, S2, and S3). The stimulus sequences in the test set were repeated ten times. The corresponding response sequences were averaged

over the repetitions. The response sequences have already been preprocessed as described in (Nishimoto et al., 2011a). Briefly, they have been realigned to compensate for motion, detrended to compensate for drift and z-scored. Additionally, the first six seconds of the blocks were discarded. No further preprocessing was performed. Regions of interests were localized using the multifocal retinotopic mapping technique on retinotopic mapping data that were acquired in separate sessions (Hansen et al., 2004). As a result, the voxels were grouped into 16 areas. However, not all areas were identified in all subjects (Table 6.1). The last 45 seconds of the blocks in the training set were used as the validation set.

## Problem statement

Let $\mathbf{x}^t \in \mathbb{R}^n$ and $\mathbf{y}^t \in \mathbb{R}^m$ be a stimulus and a response at temporal interval $[t, t+1]$, where $n$ is the number of stimulus dimensions and $m$ is the number of voxel responses. We are interested in predicting the most likely response $\mathbf{y}^t$ given the stimulus history $\mathbf{X}^t = (\mathbf{x}^0, \ldots, \mathbf{x}^t)$:

$$\widehat{\mathbf{y}}^t = \arg\max_{\mathbf{y}^t} (\Pr(\mathbf{y}^t | \mathbf{X})) \tag{6.1}$$

$$= \mathbf{g}(\phi(\mathbf{x}^0, \ldots, \phi(\mathbf{x}^t))) \tag{6.2}$$

where Pr is an encoding distribution, $\phi$ is a feature model such that $\phi(\cdot) \in \mathbb{R}^p$, $p$ is the number of feature dimensions, and $\mathbf{g}$ is a response model such that $\mathbf{g}(\cdot) \in \mathbb{R}^m$.

In order to solve this problem, we must define the feature model that transforms stimuli to features and the response model that transforms features to responses. We used two alternative feature models; a scene description model that codes for low-level visual

features (Oliva & Torralba, 2001) and a word embedding model that codes for high-level semantic content. We used two response model families that differ in architecture (recurrent neural network family and feedforward ridge regression family) (Figure 6.1). In contrast to standard convolution models for fMRI time series, we are dealing with potentially very large feature spaces. This means that in the absence of constraints the optimization of model parameters can be ill posed. Therefore, we use dropout and early stopping for the recurrent models, and $L^2$ regularization for the feedforward models.

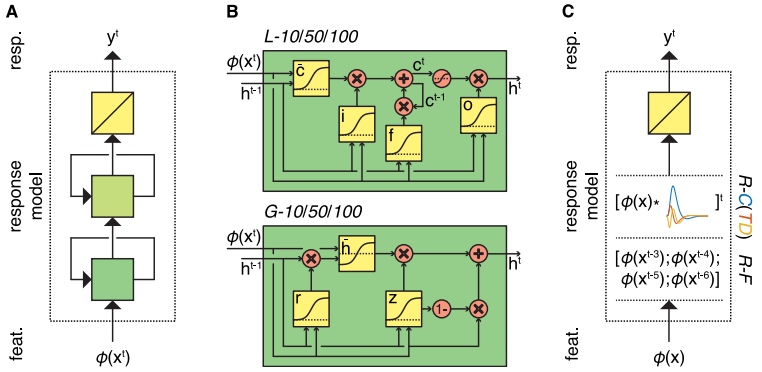



**Figure 6.1:** Overview of the response models. ***A***: Response models in the RNN family. All RNN models process feature sequences via two (recurrent) nonlinear layers and one (nonrecurrent) linear layer but differ in the type and number of artificial neurons. *L-10/50/10* models have 10, 50, or 100 long short-term memory units in both of their hidden layers, respectively. Similarly, *G-10/50/10* models have 10, 50, or 100 gated recurrent units in both of their hidden layers, respectively. ***B***: First-layer long short-term memory and gated recurrent units. Squares indicate linear combination and nonlinearity. Circles indicate elementwise operations. Gates in the units control the information flow between the time points. ***C***: Response models in the ridge regression family. All ridge regression models process feature sequences via one (nonrecurrent) linear layer but differ in how they account for the hemodynamic delay. *R-C(TD)* models convolve the feature sequence with the canonical hemodynamic response function (and its time and dispersion derivatives). *R-F* model lags the feature sequence for 3, 4, 5, and 6 s and concatenates the lagged sequences.

## Feature models

### High-level semantic model

As a high-level semantic model we used the word2vec (W2V) model by Mikolov et al. (2013a), Mikolov et al. (2013b), Mikolov et al. (2013a). This is a one-layer feedforward neural network that is trained for predicting either target words/phrases from source-context words (continuous bag-of-words) or source context-words from target words/phrases (skip-gram). Once trained, its hidden states are used as continuous distributed representations of words/phrases. These representations capture many semantic regularities. We used the pretrained (skip-gram) W2V model to avoid training from scratch (https://code.google.com/archive/p/word2vec/). It was trained on 100 billion-word Google News dataset. It contains 300-dimensional continuous distributed representations of three million words/phrases.

We used the W2V model for transforming a stimulus sequence to a feature sequence on a second-by-second basis as follows: First, each one second of the stimulus sequence is assigned 20 categories (words/phrases). We used the *Clarifai* service (http://www.clarifai.com/) to automatically assign the categories rather than annotating them by hand. *Clarifai* provides a web-based video recognition application, which internally uses a pretrained deep neural network to automatically tag the contents of the video frames on a second-by-second basis. Then, each category is transformed into continuous distributed representations of words/phrases. Next, these representations are averaged over the categories. This resulted in a 300-dimensional feature vector per second of stimulus sequence ($p = 300$).

### Low-level visual feature model

As a low-level visual feature model we used the GIST model (Oliva & Torralba, 2001). The GIST model transforms scenes into spatial envelope representations. These representations capture many perceptual dimensions that represent the dominant spatial structure of a scene and have been used to study neural representations in a number of earlier work (Groen, Ghebreab, Prins, Lamme & Scholte, 2013; Leeds, Seibert, Pyles & Tarr, 2013; Cichy et al., 2016). We used the implementation that is provided at: http://people.csail.mit.edu/torralba/code/spatialenvelope/.

We used the GIST model for transforming a stimulus sequence to a feature sequence on a second-by-second basis as follows: First, each 16 non-overlapping $8 \times 8$ regions of all 15 $128 \times 128$ frames in one second of the stimulus sequence are filtered with 32 Gabor filters that have eight orientations and four scales. Then, their energies are averaged over the frames. This resulted in a 512-dimensional feature vector per second of stimulus sequence ($p = 512$).

## Response models

### Ridge regression family

The response models in the ridge regression family predict feature-evoked responses as a linear combination of features. Each member of this family differs in how it accounts for the hemodynamic delay.

The *R-C* model (i) convolves the features with the canonical hemodynamic response function (Friston et al., 1994) and (ii) predicts the responses as a linear combination of these features:

$$\widehat{\mathbf{y}}^t = (\mathbf{H}_c\mathbf{F}_c\mathbf{B}^\top)^t \tag{6.3}$$

where $\mathbf{H}_c \in \mathbb{R}^{t\times t}$ is the Toeplitz matrix of the canonical HRF. That is, it is a diagonal-constant matrix that contains the shifted versions of the HRF in its columns. Multiplying it with a signal corresponds to convolution of the HRF with the signal. Furthermore, $\mathbf{F}_c = [\phi(\mathbf{x}^0), \ldots, \phi(\mathbf{x}^t)]^\top \in \mathbb{R}^{t\times p}$ and $\mathbf{B} \in \mathbb{R}^{m\times p}$ is the matrix of regression coefficients.

The *R-CTD* model (i) convolves the features with the canonical hemodynamic response function, its temporal derivative and its dispersion derivative (Friston, Josephs, Rees & Turner, 1998), (ii) concatenates these features and (iii) predicts the responses as a linear combination of these features:

$$\widehat{\mathbf{y}}^t = ([\mathbf{H}_c\mathbf{F}_c, \mathbf{H}_{ct}\mathbf{F}_c, \mathbf{H}_{cd}\mathbf{F}_c]\mathbf{B}^\top)^t \tag{6.4}$$

where $\mathbf{H}_{ct} \in \mathbb{R}^{t\times t}$ is the Toeplitz matrix of the the temporal derivative of the canonical HRF, $\mathbf{H}_{cd} \in \mathbb{R}^{t\times t}$ is the Toeplitz matrix of the the dispersion derivative of the canonical HRF and $\mathbf{B} \in \mathbb{R}^{m\times 3p}$ is the matrix of regression coefficients.

The *R-F* model is a finite impulse response (FIR) model that (i) lags the features for 3, 4, 5, and 6 s (Nishimoto et al., 2011a), (ii) concatenates these features and (iii) predicts the responses as a linear combination of these features:

$$\widehat{\mathbf{y}}^t = \mathbf{F}_f\mathbf{B}^\top \tag{6.5}$$

where $\mathbf{F}_f = [\phi(\mathbf{x}^{t-3}), \phi(\mathbf{x}^{t-4}), \phi(\mathbf{x}^{t-5}), \phi(\mathbf{x}^{t-6})]^\top \in \mathbb{R}^{t \times 4p}$ and $\mathbf{B} \in \mathbb{R}^{m \times 4p}$ is the matrix of regression coefficients.

We used the validation set for model selection (a regularization parameter per voxel) and the training set for model estimation (a row of $\mathbf{B}$ per voxel). Regularization parameters were selected as explained in (Güçlü & van Gerven, 2014). The rows of $\mathbf{B}$ were estimated by analytically minimizing the $L^2$-penalized least squares loss function. In related Bayesian models, this corresponds to applying shrinkage priors to the parameters (weights) of our model.

### Recurrent neural network family

The response models in the RNN family are two-layer recurrent neural network models. They use their internal memories for nonlinearly processing arbitrary feature sequences and predicting feature-evoked responses as a linear combination of their second-layer hidden states:

$$\widehat{\mathbf{y}}^t = \mathbf{h}_2^t \mathbf{W}^\top \tag{6.6}$$

where $\mathbf{h}_2^t$ represents the hidden states in the second layer, and $\mathbf{W}$ are the weights. The RNN models differ in the type and number of artificial neurons.

The *L-10, L-50*, and *L-100* models are two-layer recurrent neural networks that have 10, 50, and 100 long short-term memory (LSTM) units (Hochreiter & Schmidhuber, 1997) in their hidden layers, respectively. Each LSTM unit has a cell state that acts as its internal memory by storing information from previous time points. The contents of the cell state are modulated by the gates of

the unit and in turn modulate its outputs. As a result, the output of the unit is not only controlled by the present stimulus alone, but also by the stimulus history. The gates are implemented as multiplicative sigmoid functions of the inputs of the unit at the current time point and the outputs of the unit at the previous time point. That is, the gates produce values between zero and one, which are multiplied by (a function of) the cell state to determine the amount of information to store, forget or retrieve at each time point. The first-layer hidden states of an LSTM unit are defined as follows:

$$\mathbf{h}^t = \mathbf{o}^t \odot \tanh(\mathbf{c}^t) \tag{6.7}$$

$$\mathbf{o}^t = \sigma(\mathbf{U}_o \mathbf{h}^{t-1} + \mathbf{W}_o \phi(\mathbf{x}^t) + \mathbf{b}_o) \tag{6.8}$$

where $\odot$ denotes elementwise multiplication, $\mathbf{c}^t$ is the cell state, and $\mathbf{o}^t$ are the output gate activities. The cell state maintains information about the previous time points. The output gate controls what information will be retrieved from the cell state. The cell state of an LSTM unit is defined as:

$$\mathbf{c}^t = \mathbf{f}^t \odot \mathbf{c}^{t-1} + \mathbf{i}^t \odot \overline{\mathbf{f}}^t \tag{6.9}$$

$$\mathbf{f}^t = \sigma(\mathbf{U}_f \mathbf{h}^{t-1} + \mathbf{W}_f \phi(\mathbf{x}^t) + \mathbf{b}_f) \tag{6.10}$$

$$\mathbf{i}^t = \sigma(\mathbf{U}_i \mathbf{h}^{t-1} + \mathbf{W}_i \phi(\mathbf{x}^t) + \mathbf{b}_i) \tag{6.11}$$

$$\overline{\mathbf{f}}^t = \sigma(\mathbf{U}_c \mathbf{h}^{t-1} + \mathbf{W}_c \phi(\mathbf{x}^t) + \mathbf{b}_c) \tag{6.12}$$

where $\mathbf{f}^t$ are the forget gate activities, $\mathbf{i}^t$ are the input gate activities, and $\overline{\mathbf{f}}^t$ is an auxiliary variable. Forget gates control what old information will be discarded from the cell states. Input gates control what new information will be stored in the cell states. Furthermore, **U**s and **W**s are the weights and **b**s are the biases that

determine the behavior of the gates (i.e., the learnable parameters of the model).

The *G-10*, *G-50*, and *G-100* models are two-layer recurrent neural networks that have 10, 50, and 100 gated recurrent units (GRU) (Cho et al., 2014) in the their hidden layers, respectively. The GRU units are simpler alternatives to the LSTM units. They combine hidden states with cell states and input gates with forget gates. The first-layer hidden states of a GRU unit is defined as follows:

$$\mathbf{h}^t = (1 - \mathbf{z}^t) \odot \mathbf{h}^{t-1} + \mathbf{z}^t \odot \overline{\mathbf{h}}^t \tag{6.13}$$

$$\mathbf{z}^t = \sigma(\mathbf{U}_z \mathbf{h}^{t-1} + \mathbf{W}_z \phi(\mathbf{x}^t) + \mathbf{b}_z) \tag{6.14}$$

$$\mathbf{r}^t = \sigma(\mathbf{U}_r \mathbf{h}^{t-1} + \mathbf{W}_r \phi(\mathbf{x}^t) + \mathbf{b}_r) \tag{6.15}$$

$$\overline{\mathbf{h}}^t = \tanh(\mathbf{U}_h(\mathbf{r}^t \odot \mathbf{h}^{t-1}) + \mathbf{W}_h \sigma(\mathbf{x}^t) + \mathbf{b}_h) \tag{6.16}$$

where $\mathbf{z}^t$ are update gate activities, $\mathbf{r}^t$ are reset gate activities and $\overline{\mathbf{h}}^t$ is an auxiliary variable. Like the gates in LSTM units, those in GRU units control the information flow between the time points. As before, **U**s and **W**s are the weights and **b**s are the biases that determine the behavior of the gates (i.e., the learnable parameters of the model).

The second-layer hidden states are defined similarly to the first-layer hidden states except for replacing the input features with the first-layer hidden states. For each previously identified brain area of each subject, a separate model was trained. That is, the voxels in a given brain area of a given subject shared the same recurrent layers but had different weights for linearly transforming the hidden states of the second recurrent layer to the response predictions. We used truncated backpropagation through time in conjunction with the optimization method Adam (Kingma & Ba, 2014) to train the models on the training set by iteratively min-

imizing the mean squared error loss function. Dropout (Hinton et al., 2012) was used to regularize the hidden layers. The epoch in which the validation performance was the highest was taken as the best model. The *Chainer* framework (http://chainer.org/) was used to implement the models.

## HRF estimation

Voxel-specific HRFs were estimated by stimulating the RNN model with an impulse. Let $\mathbf{x}^{-t}, \ldots, \mathbf{x}^{0}, \ldots, \mathbf{x}^{t}$ be an impulse such that $\mathbf{x}$ is a vector of zeros at times other than time 0 and a vector of ones at time 0. The period of the impulse before time 0 is used to stabilize the baseline of the impulse response. First, the response of the model to the impulse is simulated:

$$[\mathbf{H}_r^*]_{-t}^{t} = \mathbf{g}_r(\mathbf{x}^{-t}, \ldots, \mathbf{x}^{0}, \ldots, \mathbf{x}^{t}) \tag{6.17}$$

where $[\mathbf{H}_r^*]_{-t}^{t} = (\mathbf{H}_r^{*-t}, \ldots, \mathbf{H}_r^{*0}, \ldots, \mathbf{H}_r^{*t})$. Then, the baseline of the impulse response before time 0 is subtracted from itself:

$$[\mathbf{H}_r^*]_{-t}^{t} = [\mathbf{H}_r^*]_{-t}^{t} - \mathbf{H}_r^{*-1} \tag{6.18}$$

Next, the impulse response is divided by its maximum:

$$[\mathbf{H}_r^*]_{-t}^{t} = [\mathbf{H}_r^*]_{-t}^{t} / \max [\mathbf{H}_r^*]_{-t}^{t} \tag{6.19}$$

Finally, the period of the impulse response before time 0 is discarded, and the remaining period of the impulse response is taken as the HRF of the voxels:

$$[\mathbf{H}_r]_0^t = [\mathbf{H}_r^*]_0^t \tag{6.20}$$

The time when the HRF is at its maximum was taken as the delay of the response, and the time after the delay of the response when the HRF was at its minimum was taken as the delay of undershoot.

## Performance assessment

The performance of a model for a voxel was defined as the cross-validated Pearson's product-moment correlation coefficient between the observed and predicted responses of the voxel [1]. Its performance for a group of voxels was defined as the median of its performance over the voxels in the group (r∼). The data of all subjects were concatenated prior to analyzing the performance of the models.

In order to make sure that the differences in the performance of a model in different areas are not caused by the differences in the signal-to-noise ratios of the areas, the performance of the model in an area was corrected for the median of the noise ceilings of the voxels in the area ($\widetilde{r}^*$) (Kay et al., 2013a). Briefly, we performed Monte Carlo simulations in which the correlation coefficient between a signal and a noisy signal is estimated. In each simulation, both the signal and the noise were drawn from a Gaussian distribution. The noisy signal was taken to be the

[1] The cross-validated correlation coefficient automatically penalizes for model complexity and therefore can be used as a proxy for model evidence.

summation of the signal sample and the noise sample. The parameters of the signal and the noise distributions were estimated from the 10 repeated measurements of the responses to the same stimuli. The noise distribution was assumed to be zero mean, and its variance was taken to be the variance of the standard errors of the data. The mean and the variance of the signal distribution were given as the mean of the data, and the difference between the variance of the data and the noise distribution, respectively. The medians of the correlation coefficients that were estimated in the simulations were taken to be the noise ceilings of the voxels, indicating the maximum performance that can be expected from the perfect model due to the noise in the data.

Permutation tests were used for comparing the performance of a model against chance level. First, data were randomly permuted over time for 200 times. Then, a separate model was trained and tested for each of the 200 permutations. Finally, the *p*-value was taken to be the fraction of the 200 permutations whose performance was greater than the actual performance. The performance was considered significant at $\alpha = 0.05$ if the *p*-value was less than 0.05 (Bonferroni corrected for number of areas).

Bootstrapping was used for comparing the performance of two models over voxels in a ROI (i.e., all voxels or voxels in an area). For 10,000 repetitions, bootstrap samples (i.e., voxels) were drawn from the ROI with replacement, and the performance difference between the models over these voxels were estimated. The performance difference was considered significant at $\alpha = 0.05$ if the 95% confidence interval of the sampled statistic did not cover zero (Bonferroni corrected for number of models).

## 6.3 Results

## Comparison of response models

We evaluated the response models by comparing the performance of the response models in the (recurrent) RNN family and (feed-forward) ridge regression family in combination with the (high-level) W2V model and the (low-level) GIST model. Using two feature models of different levels ruled out any potential biases in the performance difference of the response models that can be caused by the feature models. Recall that the models in the RNN family (*G/L-10/50/100* models) differed in the type and number of artificial neurons, whereas the models in the ridge regression family (*R-C/R-CTD/R-F* models) differed in how they account for the hemodynamic delay.

Once the best response models among the RNN family and the ridge regression family were identified, we first compared their performance in detail. Particular attention was paid to the voxels where the performance of the models differed by more than an arbitrary threshold of $r = 0.1$. We then compared the performance of the best response model among the RNN family over the areas along the visual pathway.

### Comparison of the response models in combination with the semantic model

Figure 6.2 compares the performance of all response models in combination with the W2V model. The performance of the models in the RNN family that had 50 or 100 artificial neurons was always significantly higher than that of all models in the ridge regression family ($p \leq 0.05$, bootstrapping). However, the performance of the models in the same family was not always significantly different from each other. The performance of the *G-100* model was the highest among the RNN family ($\widetilde{r} = 0.16$), and that of

the *R-C* model was the highest among the ridge regression family ($\widetilde{r} = 0.12$).

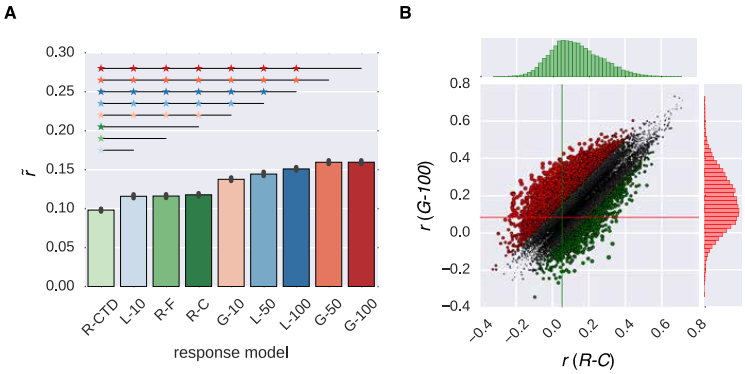



**Figure 6.2:** Comparison of the response models in combination with the W2V model. ***A***: Median performance of response models in RNN (*G-X* and *L-X*) and ridge regression (*R-X*) families over all voxels. Error bars indicate 95% confidence intervals (bootstrapping). Asterisks indicate significant performance difference. All of the individual bars depict significantly above chance-level performance ($p < 0.05$, permutation test). ***B***: Performance of best response models in RNN (*G-100* model) and ridge regression (*R-C* model) families over individual voxels. Points indicate voxels. Gray points indicate voxels where the performance difference is less than $r = 0.1$. Lines indicate (median) performance over all voxels.

The performance of the *G-100* model and the *R-C* model differed from each other by more than the chosen threshold of $r = 0.1$ in 30% of the voxels. The performance of the *G-100* model was higher in 78% of these voxels ($\Delta\widetilde{r} = 0.17$), and that of the *R-C* model was higher in 22% of these voxels ($\Delta\widetilde{r} = 0.14$).

Figure 6.3 compares the performance of the *G-100* model in combination with the W2V model over the areas along the visual stream. While the performance of the model was significantly higher than chance throughout the areas ($p \leq 0.05$, permutation test), it was particularly high in downstream areas. For example, it was the highest in TOS ($\widetilde{r}^* = 0.55$), OFA ($\widetilde{r}^* = 0.38$) and EBA

($\widetilde{r}^* = 0.35$), and the lowest in pSTS ($\widetilde{r}^* = 0.14$), IPS ($\widetilde{r}^* = 0.20$) and V1 ($\widetilde{r}^* = 0.24$).

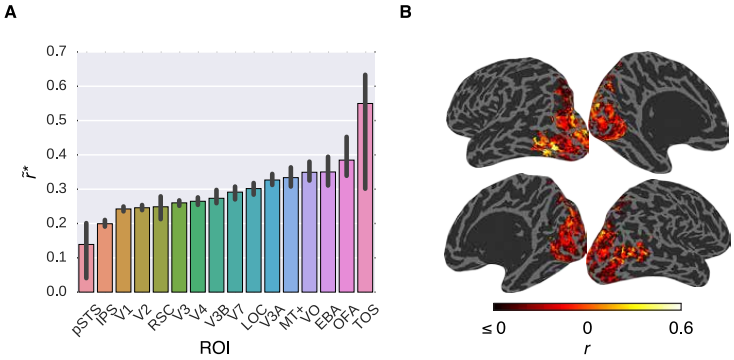



**Figure 6.3:** Comparison of the ***G-100*** model in combination with the W2V model in different areas. ***A***: Median noise ceiling controlled performance over all voxels in different areas. Error bars indicate 95% confidence intervals (bootstrapping). All of the individual bars depict significantly above chance-level performance ($p < 0.05$, permutation test). ***B***: Projection of performance to cortical surfaces of S3.

### Comparison of the response models in combination with the low-level feature model

Figure 6.4 compares the performance of the all response models in combination with the GIST model. The trends that were observed in this figure were similar to those that were observed in Figure 6.2. The *G-100* model was the best among the RNN family ($\widetilde{r} = 0.18$), and the *R-C* model was the best among the ridge regression family ($\widetilde{r} = 0.14$).

The *G-100* model and the *R-C* differed from each other by more than the threshold of $r = 0.1$ in 27% of the voxels. The *G-100* model was better in 66% of these voxels ($\Delta\widetilde{r} = 0.17$). The *R-C* model was better in 34% of these voxels ($\Delta\widetilde{r} = 0.14$).

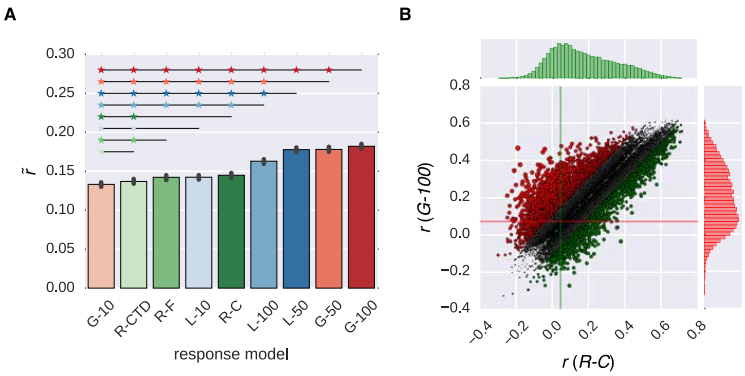



**Figure 6.4:** Comparison of the response models in combination with the GIST model. ***A***: Median performance of response models in RNN (*G-X* and *L-X*) and ridge regression (*R-X*) families over all voxels. Error bars indicate 95% confidence intervals (bootstrapping). Asterisks indicate significant performance difference. All of the individual bars depict significantly above chance-level performance ($p < 0.05$, permutation test). ***B***: Performance of best response models in RNN (*G-100* model) and ridge regression (*R-C model*) families over individual voxels. Points indicate voxels. Gray points indicate voxels where the performance difference is less than $r = 0.1$. Lines indicate median performance over all voxels.

Figure 6.5 compares the performance of the *G-100* model in combination with the GIST model over the areas along the visual pathway. While the *G-100* model performed significantly better than chance throughout the areas ($p \leq 0.05$, permutation test), it performed particularly well in upstream visual areas. For example, it performed the best in V1 ($\widetilde{r}^* = 0.39$), V2 ($\widetilde{r}^* = 0.35$) and V3 ($\widetilde{r}^* = 0.35$), and the worst in TOS ($\widetilde{r}^* = 0.13$), IPS ($\widetilde{r}^* = 0.16$) and pSTS ($\widetilde{r}^* = 0.16$).

## Comparison of feature models

Once the efficacy of the proposed RNN models was positively assessed, we performed a validation experiment in which we assessed the extent to which these models can replicate the earlier

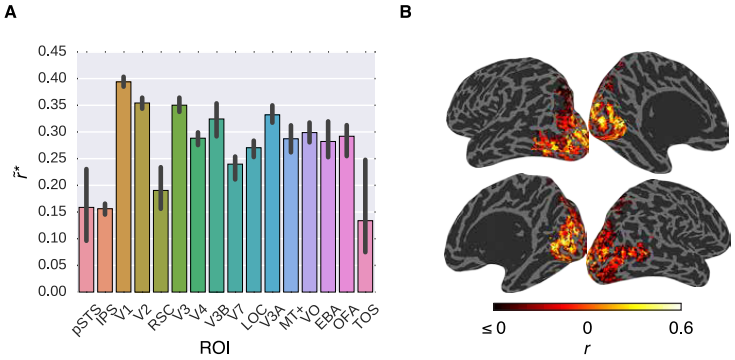



**Figure 6.5:** Comparison of the ***G-100*** model in combination with the GIST model in different areas. ***A***: Median noise ceiling controlled performance over all voxels in different areas. Error bars indicate 95% confidence intervals (bootstrapping). All of the individual bars depict significantly above chance-level performance ($p < 0.05$, permutation test). ***B***: Projection of performance to cortical surfaces of S3.

findings on the low-level and high-level subdivision of the visual cortex. This was accomplished by identifying the voxels that prefer semantic representations vs. low-level representations. Concretely, we compared the performance of the W2V model and the GIST model in combination with the *G-100* model (Figure 6.6).

The performance of the models was significantly different in all areas along the visual stream except for pSTS and V3A ($p \leq 0.05$, bootstrapping). This difference was in favor of semantic representations in downstream areas and low-level representations in upstream areas. The largest difference in favor of semantic representations was in TOS ($\Delta\widetilde{r} = 0.11$), OFA ($\Delta\widetilde{r} = 0.08$) and MT+ ($\Delta\widetilde{r} = 0.04$), and low-level representations was in V1 ($\Delta\widetilde{r} = 0.10$), V2 ($\Delta\widetilde{r} = 0.07$) and V3 ($\Delta\widetilde{r} = 0.05$).

Thirty-nine percent of the voxels preferred either representation by more than the arbitrary threshold of $r = 0.1$. Thirty-four percent of these voxels preferred semantic representations ($\Delta\widetilde{r} = 0.16$), and

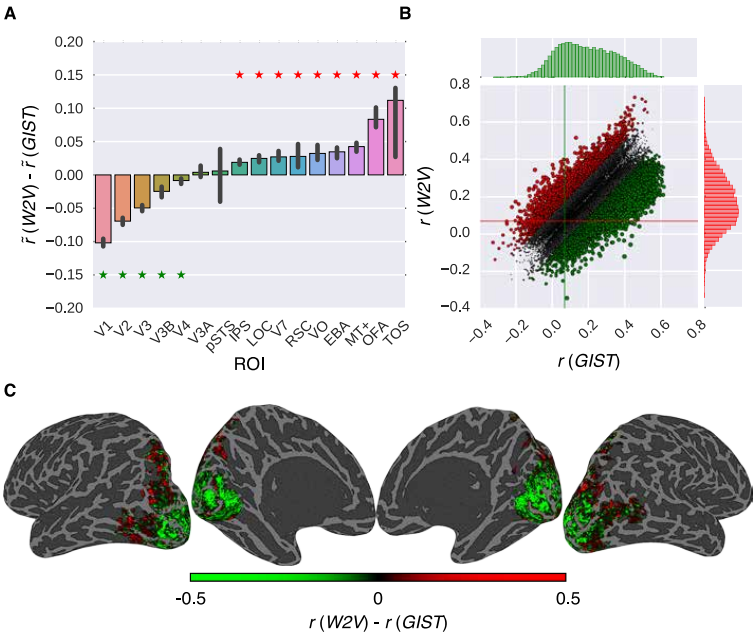



**Figure 6.6:** Comparison of the feature models in combination with the ***G-100*** model. ***A***: Median performance difference over all voxels in different areas. Asterisks indicate significant performance difference. Error bars indicate 95% confidence intervals (bootstrapping). ***B***: Performance over individual voxels. Points indicate voxels. Gray points indicate voxels where performance difference is less than $r = 0.1$. Lines indicate median performance over all voxels. ***C***: Projection of performance difference to cortical surfaces of S3.

66% percent of these voxels preferred low-level representations ($\Delta\widetilde{r} = 0.18$).

These results are in line with a large number of earlier work that showed similar dissociations between the representations of the upstream and downstream visual areas (Mishkin et al., 1983; Naselaris et al., 2009; DiCarlo, Zoccolan & Rust, 2012; Güçlü & van Gerven, 2015).

## Analysis of internal representations

Next, to gain insight into the temporal dependencies captured by the *G-100* model, we analyzed its internal representations (Figure 6.7).

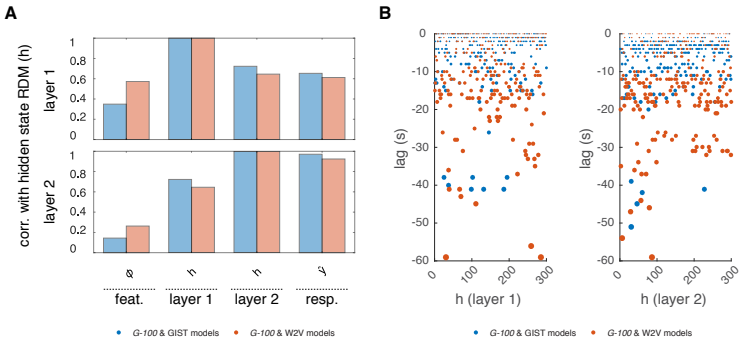



**Figure 6.7:** Internal representations of the ***G-100*** model. ***A***: Correlation between representational dissimilarity matrices of layer 1 and layer 2 hidden states with each other as well as with those of features and predicted responses. ***B***: Temporal selectivity of layer 1 and layer 2 hidden units. Points indicate lags at which cross-correlations between hidden states and features are highest.

First, we investigated how the hidden states of the RNN depend on its inputs and output. We constructed representational dissimilarity matrices (RDMs) of the stimulus sequence in the test set at different stages of the processing pipeline and averaged them over subjects (Kriegeskorte, 2008). Per feature model, this resulted in one RDM for the features, two RDMs for the layer 1 and layer 2 hidden states and one RDM for the predicted responses. We correlated the upper triangular parts of the RDMs with one another, which resulted in a value indicating how much the hidden states of the RNN were modulated by its inputs and how much they modulated its outputs at a given time point. We found a gradual increase in correlations of the RDMs. That is, the RDMs at each stage were more correlated with those at the next stage compared to those at the previous stages. Importantly, the hidden

state RDMs were highly correlated with the predicted response RDMs ($r = 0.61$ and $r = 0.93$ for layers 1 and 2, respectively) but less so with the feature RDMs ($r = 0.39$ and $r = 0.21$ for layers 1 and 2, respectively). This means that while the hidden states of the RNN modulated its outputs at a given time point, they were not modulated by its inputs to the same extent at the same time point. This suggests that a substantial part of the output at a given time-point is not directly related to the input at the same time-point, but instead to previous time-points. That is, the RNN learned to use the input history to make its predictions as expected.

Then, we investigated which time points in the input history were used by the RNN to make its predictions. We cross-correlated each hidden state with each stimulus feature, and averaged the cross-correlations over the features, which resulted in a value indicating how much a hidden state is selective to different time points in the input history. The time point at which this value was at its maximum was taken as the optimal lag of that hidden unit. We found that different hidden units had different optimal lags. The majority of the hidden units had optimal lags up to -20 s, which are likely capturing the hemodynamic factors. However, there was a non-negligible number of hidden units with optimal lags beyond this period, which might be capturing other cognitive/neuronal factors or factors related to stimulus/feature statistics. It should be noted that not all hidden units, in particular those with extensive lags, can be attributed to any of these factors, and their behavior might be induced by model definition or estimation. Furthermore, the optimal lags of the hidden units in the *W2V* based model were on average significantly higher than those in the *GIST* based model ($\mu$ = -9.6 s vs. $\mu$ = -4.9 s, $p < 0.05$, two-sample *t*-test), which might reflect the differences in the statistics of the features that the models are based on. That is, high-level semantic features tend to be more persistent than the low-level structural features across the input sequence. For example, over a given

video sequence, distribution of objects in a scene change relatively slowly compared to that of the edges in the scene.

## Estimation of voxel-specific HRFs

Traditionally, models have used analytically derived (Friston et al., 1998) or statistically estimated (Dale, 1999; Glover, 1999) HRFs such as the linear models considered here. Estimation of voxel-specific HRFs is an important problem since using the same HRF for all voxels ignores the variability of the hemodynamic response across the brain, which might adversely affect the model performance. Recent developments have focused on the derivation and estimation of more accurate HRFs. For example, Aquino et al. (2014) has shown that HRFs can be analytically derived from physiology, and Pedregosa, Eickenberg, Ciuciu, Thirion and Gramfort (2015) has shown that HRFs can be efficiently estimated from data. Note that, while the methods for statistically estimating HRFs are particularly suited for use in block designs and event related designs, they are less straightforward to use in continuous designs such as the one considered here.

As demonstrated in the previous subsection, one important advantage of the response models in the RNN family is that they can capture certain temporal dependencies in the data, which might correspond to the HRFs of voxels. Here, we evaluate the voxel-specific HRFs that are obtained by stimulating the *G-100* model with an impulse. We used both feature models in combination with the *G-100* model to estimate the HRFs of the voxels where the performance of any model combination was significantly higher than chance (51% of the voxels, $p \leq 0.05$, Student's *t*-test, Bonferroni correction) (Figure 6.8). The W2V and *G-100* models were used to estimate the HRFs of the voxels where their performance was higher than that of the GIST and *G-100* models, and vice versa.

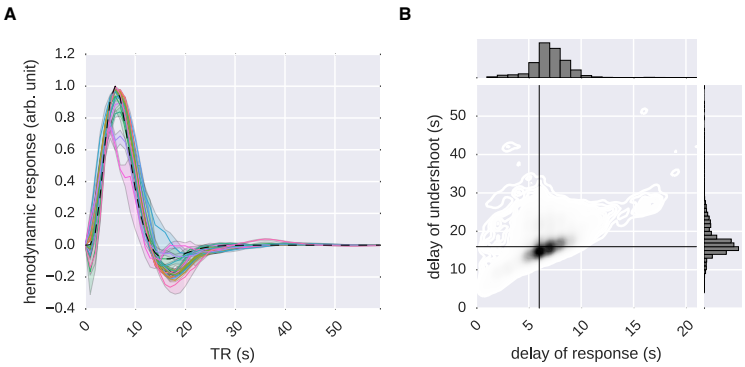



**Figure 6.8:** Estimation of the hemodynamic response functions. The *G-100* model was stimulated with an impulse. The impulse response was processed by normalizing its baseline and scale. The result was taken as the HRF. ***A***: Median hemodynamic response functions of all voxels in different areas. Error bands indicate 68% confidence intervals (bootstrapping). Different colors indicate different areas. Dashed line indicates canonical hemodynamic response function. ***B***: Delays of responses and undershoots of all voxels. Black lines indicate canonical delays.

It was found that the global shape of the estimated HRFs was similar to that of the canonical HRF. However, there was a considerable spread in the estimated delays of responses and the delays of undershoots (median delay of response = 6.57 ± 0.02 s, median delay of undershoot = 16.95 ± 0.04 s), with the delays of responses being significantly correlated with the delays of undershoots (Pearson's $r = 0.45$, $p \leq 0.05$, Student's $t$-test).

These results demonstrate that RNNs can not only learn (stimulus) feature-response relationships but also can estimate HRFs of voxels, which in turn demonstrate that the nonlinear temporal dynamics that are learned by the RNNs capture biologically relevant temporal dependencies. Furthermore, the variability in the estimated voxel-specific HRFs revealed by the recurrent models might provide a partial explanation of the performance difference between the recurrent and ridge regression models since the ridge

regression models use fixed or restricted HRFs, making it difficult for them to take such variability into account.

## 6.4 Discussion

Understanding how the human brain responds to its environment is a key objective in neuroscience. This study has shown that recurrent neural networks are exquisitely capable of capturing how brain responses are induced by sensory stimulation, outperforming established approaches augmented with ridge regression. This increased sensitivity has important consequences for future studies in this area.

### Testing hypotheses about brain function

Like any other encoding model, RNN based encoding models can be used to test hypotheses about neural representations (Naselaris et al., 2011). That is, they can be used to test whether a particular feature model outperforms alternative feature models when it comes to explaining observed data. As such, we have shown that a low-level visual feature model explains responses in upstream visual areas well, whereas a high-level semantic model explains responses in downstream visual areas well, conforming to the well established early and high-level subdivision of the visual cortex (Mishkin et al., 1983; Naselaris et al., 2009; DiCarlo et al., 2012; Güçlü & van Gerven, 2015).

Furthermore, RNN-based encoding models can also be used to test hypotheses about the temporal dependencies between features and responses. For example, by constraining the temporal memory capacities of the RNN units, one can identify the optimal scale of the temporal dependencies that different brain regions are selective to.

Here, we used RNNs as response models in an encoding framework. That is, they were used to predict responses to features that were extracted from stimuli with separate feature models. However, use cases of RNNs are not limited to this setting. For example, RNN models can be used as feature models instead of response models in the encoding framework. Like CNNs, RNNs are being used to solve various problems in fields ranging from computer vision (Gregor, Danihelka, Graves & Wierstra, 2015) to computational linguistics (Zaremba, Sutskever & Vinyals, 2014). Internal representations of task-optimized CNNs were shown to correspond to neural representations in different brain regions (Kriegeskorte, 2015; Yamins & DiCarlo, 2016a). It would be interesting to see if the internal representations of task-based RNNs have similar correlates in the brain. For example, it was recently shown that RNNs develop representations that are reminiscent of their biological counterparts when they learn to solve a spatial navigation task (Kanitscheider & Fiete, 2016). Such representations may turn out to be predictive of brain responses recorded during similar tasks.

## Limitations of RNNs for investigating neural representations

RNNs can process arbitrary input sequences in theory. However, they have an important limitation in practice. Like any other contemporary neural network architecture, typical RNN architectures have a very large number of free parameters. Therefore, a very large amount of training data is required for accurately estimating RNN models without overfitting. While there are several methods to combat overfitting in RNNs like different variants of dropout (Hinton et al., 2012; Zaremba et al., 2014; Semeniuta, Severyn & Barth, 2016), it is still an important issue to which particular attention needs to be paid.

This can also be the reason why gated recurrent unit architectures were shown to outperform LSTM architectures. That is, the performance difference between the two types of architectures is likely to be caused by difficulties in model estimation in the current data regime rather than one architecture being better suited to the problem at hand than the other.

This also means that RNN models will face difficulties when trying to predict responses to very high-dimensional stimulus features such as the internal representations of convolutional neural networks which range from thousands to hundreds of thousands dimensions. For such features, dimensionality reduction techniques can be utilized for reducing the feature dimensionality to a range that can be handled with RNNs in scenarios with either insufficient computational resources or training data.

Linear response models have been used with great success in the past for gaining insights into neural representations. They have been particularly useful since linear mappings make it easy to interpret factors driving response predictions. One might argue that the nonlinearities introduced by RNNs make the interpretation harder compared to linear mappings. However, the relative difficulty of interpretation is a direct consequence of more accurate response predictions, which can be beneficial in certain scenarios. For example, it was shown that systematic nonlinearities that are not taken into account by linear mappings can lead to less accurate response predictions and tuning functions of V1 voxels (Vu et al., 2011). Furthermore, since more accurate response predictions lead to higher statistical power, the improved model fit afforded by RNNs might make detection of more subtle effects possible. Moreover, when the goal is to compare different feature models, such as the GIST and W2V models used here, maximizing explained variance might become the main criterion of interest. That is, linear models might lead to misleading performance differences between the encoding models in the cases where their assumptions about the underlying temporal dynamics

do not hold. In such cases, it would be particularly important to fit the response models as accurately as possible as to ensure that the observed performance difference between two encoding models is driven by their underlying feature representations and not suboptimal model fits. Therefore, RNNs will be particularly useful in settings where temporal dynamics are of primary interest. Finally, combining the present work with recent developments on understanding RNN representations (Karpathy, Johnson & Li, 2015) is expected to improve the interpretations of factors driving response predictions.

## Capturing temporal dependencies

RNNs can use their internal memories to capture the temporal dependencies in data. In the context of modeling the dynamics of brain activity in response to naturalistic stimuli, these dependencies can be caused by factors such as neurovascular coupling or stimulus-induced cognitive processes. By providing an RNN with an impulse on the input side, it was shown that, effectively, the RNN learns to represent voxel-specific hemodynamic responses. Importantly, the RNNs allowed us to estimate these HRFs from data collected under a continuous design. To the best of our knowledge this is the first time it has been shown that this is possible in practice. By analyzing the internal representations of an RNN, it was also shown that the RNN learns to represent information from stimulus features at past time points beyond the range of neurovascular coupling. Hence, the predictions of observed brain responses are likely induced by stimulus-related, cognitive or neural factors on top of the hemodynamic response.

## Isolating neural and haemodynamic components

In the introduction, we motivated the use of RNNs as a generic parameterization of any non-linear convolution of stimulus features to hemodynamic responses. Crucially, this could cover both neuronal and hemodynamic convolution. In other words, our black box approach allows for a neuronal convolution of stimulus feature input to produce a neuronal response that is subsequently convolved by hemodynamic operators to produce the observed outcome. This facility may explain the increased cross-validation accuracy observed in our analyses (over and above more restricted models of hemodynamic convolution). In other words, the procedure detailed in this paper can accommodate neuronal convolutions that may be precluded in conventional models.

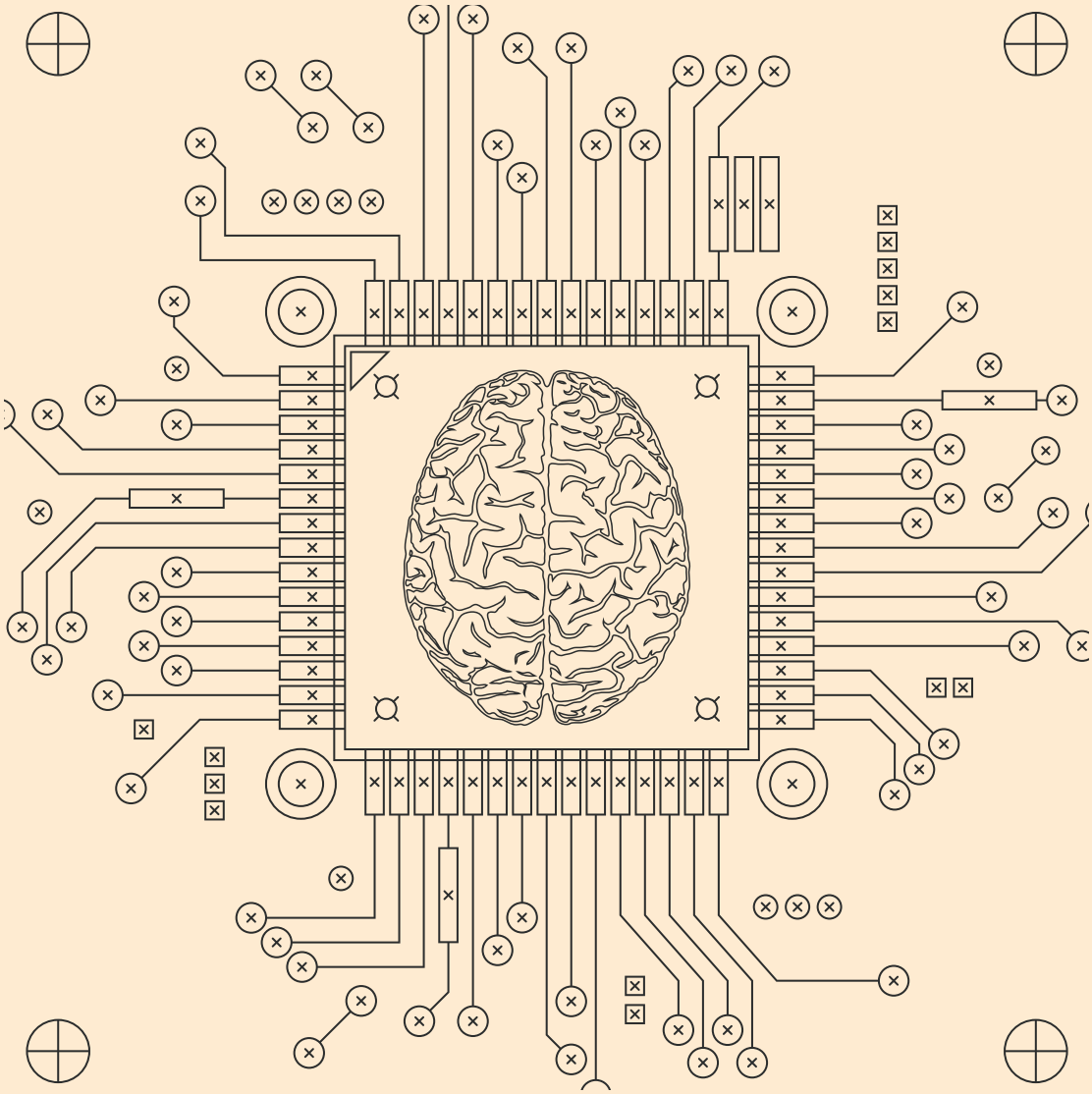
The cost of this flexibility is that we cannot separate the neuronal and hemodynamic components of the convolution. This follows from the fact that the RNN parameterization does not make an explicit distinction between neuronal and hemodynamic processes. To properly understand the relative contribution of these formally distinct processes, one would have to use a generative model approach with biologically plausible prior constraints on the neuronal and hemodynamic parts of the convolution. This is precisely the objective of dynamic causal modeling that equips a system of neuronal dynamics (and implicit recurrent connectivity) with a hemodynamic model based upon known biophysics (Friston, Harrison & Penny, 2003). It would therefore be interesting to examine the form of RNNs in relation to existing dynamic causal models that have a similar architecture.

## Conclusions

We have shown for the first time that RNNs can be used to predict how the human brain processes sensory information. Whereas classical connectionist research has focused on the use of RNNs as models of cognitive processing (Elman, 1993), the present work has shown that RNNs can also be used to probe the hemodynamic correlates of ongoing cognitive processes induced by dynamically changing naturalistic sensory stimuli. The ability of RNNs to learn about long-range temporal dependencies provides the flexibility to couple ongoing sensory stimuli that induce various cognitive processes with delayed measurements of brain activity that depend on such processes. This end-to-end training approach can be applied to any neuroscientific experiment in which sensory inputs are coupled to observed neural responses.

**Table 6.1:** Number of voxels per subject and area.

| | V2 | V3 | V1 | IPS | V4 | LOC | V7 | MT+ | V3A | V3B | VO | EBA | OFA | RSC | pSTS | TOS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| S1 | 1477 | 1141 | 994 | 2251 | 734 | 885 | 0 | 466 | 252 | 256 | 410 | 0 | 0 | 71 | 45 | 0 |
| S2 | 1659 | 1360 | 1043 | 0 | 1032 | 614 | 400 | 174 | 337 | 223 | 267 | 319 | 246 | 128 | 0 | 0 |
| S3 | 1377 | 1131 | 1366 | 893 | 750 | 408 | 583 | 263 | 282 | 225 | 0 | 131 | 91 | 8 | 16 | 41 |



# 7 Summary

# 7.1 Summary

**Chapter 2** Encoding and decoding in functional magnetic resonance imaging have recently emerged as an established area of research to noninvasively characterize the relationship between stimulus features and human brain activity. Conventionally, stimulus features are "hand-designed" by theorists or experimentalists (e.g., Gabor wavelets or semantic contents). However, hand-designing stimulus features is a slow and laborious process that is prone to the influence of subjective expectations and restricted to a priori hypotheses. Therefore, formalizing what stimulus features modulate human brain activity remains a notoriously challenging endeavor. In Chapter 2, to overcome this challenge, we introduced a general framework for making directly testable predictions of single voxel responses to statistically adapted representations of ecologically valid stimuli, which show what visual cortical representations would be like if the brain were adapted to the statistical regularities in the environment.

We developed a parsimonious computational model, which comprises two main components: (i) A sparse coding (SC) model of how early visual cortical representations are adapted to statistical regularities in natural images. This model transforms raw stimuli to stimulus features and learns the transformation from unlabeled data without any supervisory signals (Hyvärinen & Hoyer, 2001). (ii) A scalable regularized linear regression model of how populations of these representations are pooled by single voxels. This model transforms the stimulus features to voxel responses and learns the transformation from feature-transformed stimulus-response pairs. We established a baseline using the Gabor wavelet pyramid (GWP) model of phase-invariant complex cells (Kay et al., 2008).

We first estimated and validated the models using an fMRI data set of early visual voxel responses to natural images (Kay et al.,

2008; Naselaris et al., 2009; Lescroart et al., 2011). As a result, we showed that the SC model learned topographically organized, spatially localized, oriented and bandpass simple and complex cell receptive fields that were similar to those found in the primary visual cortex. Most adjacent simple and complex cells were selective to a similar location, orientation, and wavelength, whereas they were selective to a different phase. Like the simple cells, most complex cells were selective to orientation and wavelength. Unlike the simple cells, most complex cells were more phase-invariant and had a degree of spatial invariance. We then predicted single voxel responses to natural images and identified natural images from stimulus-evoked multiple voxel responses. As a result, we showed that the performance of the SC model was significantly higher than that of the GWP model. Encoding performance was quantified as the coefficient of determination between the observed and predicted single voxel responses to 120 validation images. Decoding performance was quantified as the accuracy of identifying 120 validation images in a set of 9264 candidate images (Fei-Fei et al., 2007) from stimulus-evoked multiple voxel responses. The SC model was also more invariant to translations in images than the GWP model. The performance difference between the models increased after translating the images in the validation set by five pixels (i.e. 0.8°) in a random dimension.

In summary, the key outcomes of this chapter were the following:

- Statistically adapted low-level sparse and invariant representations of natural images (i) reproduce salient features of the columnar organization of the primary visual cortex and (ii) better span the space of early visual cortical representations and can be more effectively exploited in identification than Gabor wavelets – the fundamental building blocks of many state-of-the-art encoding and decoding models.

- In contrast to hand-designing stimulus features, our framework is particularly suited to address what mid- or high-level representations (e.g. higher order statistical features or semantic contents of natural images) span the space of extrastriate visual cortical (e.g. V4 or inferior temporal gyrus) representations using highly nonlinear multi-layer statistical generative models of natural images that learn hierarchical representations and predict unknown visual cortical representations, based on different coding principles.

**Chapter 3** In Chapter 3, we started to tackle the question of what information is represented in the sensory pathways of the human brain with deep neural networks following the results of the previous chapter that suggested such models are particularly suited to probe mid- and high-level neural representations in the ventral stream.

We analyzed the same dataset as the one in Chapter 2 (natural image-human ventral stream fMRI (Kay et al., 2008; Naselaris et al., 2009; Lescroart et al., 2011) with deep encoding models. These models comprised an object recognition-optimized spatial DNN (Chatfield et al., 2014) (feature space) and ridge regression (forward model).

We also did three types of control analyses with a different feature space: (i) DNNs that had different architectures but the same task as the main DNN. (ii) DNNs that had the same architecture as the main DNN but no task. (iii) Gabor wavelet pyramid (Kay et al., 2008).

We first used the encoding models for predicting voxel responses to natural images and assigned the DNN layers to the voxels based on their encoding performance. As a result, we showed that they achieved state-of-the-art encoding performance. The properties (Kolmogorov complexity, receptive field, and spatial invariance) of the voxels that were assigned to different layers

systematically changed as a function of layer assignment and the layer assignments of the voxels systematically increased as a function of location. The low-level areas mostly had the least complex and invariant blob, contrast, and edge receptive fields. The complex and invariant contour, shape, and texture receptive fields were in the intermediate-level areas. The high-level areas mostly had the most complex and invariant irregular pattern, object part and entire object receptive fields. These results were also reproduced with the first type of control models (different architecture and same task) but neither the second nor the third type of control models (neither same architecture and no task nor Gabor wavelet pyramid). We then converted the encoding models to a decoding model for identifying natural images from stimulus-evoked human ventral stream voxel responses and showed that it could achieve state-of-the-art decoding performance.

In summary, the key outcomes of this chapter were the following:

- There exists a representational gradient such that the representations of the increasingly deeper object recognition-optimized spatial DNN layers better correspond to the neural representations of the increasingly downstream areas in the human ventral stream.

- This correspondence is driven by task-optimization and not exact architectural assumptions.

**Chapter 4** In Chapter 4, extended the ideas from the previous chapter to action recognition and dorsal stream.

As the dataset, we used a natural movie-human dorsal stream fMRI dataset (Nishimoto et al., 2011a; Nishimoto et al., 2014) instead of a natural image-human ventral stream fMRI dataset. As the model, we used a three-dimensional (spatiotemporal convolutions) convolutional deep network after finetuning it for ac-

tion recognition (in movies) (Tran et al., 2014) instead of a two-dimensional (spatial convolutions) deep neural network after it has been pretrained for object recognition (in photographs).

We did two types of analyses: (i) We reproduced the main analyses and the results from the previous chapter. (ii) We repeated these analyses on the hyperaligned fMRI data (Haxby et al., 2011) and showed the existence of a common representational space, which can be used for within subject analyses.

In summary, the key outcomes of this chapter were the following:

- Deep neural network and dorsal stream representations show correspondence.
- Dorsal stream representations are shared between subjects.
- A common encoder can predict fMRI responses to novel stimuli for unseen subjects.
- A common decoder can identify novel stimuli from fMRI responses for unseen subjects.

**Chapter 5** In Chapter 5, we continued to tackle the question of what information is represented in the sensory pathways of the human brain. In the previous two chapters, we showed that representations of task-optimized visual deep neural network layers correspond to neural representations in the visual pathways. In this chapter, we looked at whether this correspondence holds for task-optimized auditory deep neural network layers and neural representations in the auditory pathway or not.

As the datasets, we used the MagnaTagATune dataset (~25k song excerpts, 188/50 tags) (Law et al., 2009) and the Study-Forrest dataset (25 song excerpts, 20/12 subject fMRI measure-

ments) (Hanke et al., 2015). As the main models, we used three one-dimensional AlexNet (Krizhevsky et al., 2012) variants: (i) DNN-T model (time domain), (ii) DNN-F model (frequency domain) and (iii) DNN-TF model (time and frequency domains). As the control models, we used four standard auditory models: (i) MFS, (ii) MFCC, (iii) low-quefrency MFS and (iv) high-quefrency MFS (Hanke et al., 2015). As another control model, we used a random (untrained) DNN-TF model.

We first performed automatic music tagging analysis. We evaluated the performance of the main models over all tags and the performance of the task optimized DNN-TF model over individual tags and showed that the main models achieved state-of-the-art automatic music tagging performance. We then performed representational similarity analysis (Kriegeskorte, 2008). We compared the global STG RDM with the main and control model RDMs (region of interest analysis). As a result, we showed that the task-optimized DNN-TF model captured the representational geometry of STG. However, the control models failed to do so. We also compared the local STG RDMs with the task-optimized and random DNN-TF model RDMs (searchlight analysis) and assigned the DNN-TF model layers to the STG voxels based on their representational similarity. As a result, we showed that the task-optimized DNN-TF model revealed a representational gradient in STG: the shallow and deep task-optimized DNN-TF model layers were assigned to the anterior and posterior STG voxels, respectively. However, the random DNN-TF model again failed to do so.

In summary, the key outcomes of this chapter were the following:

- Increasingly downstream neural representations in STG can be modeled with representations of increasingly deeper task-optimized auditory DNN layers.

- Task optimization is important for modeling neural representations in STG.

- Results are in line with visual neuroscience literature.

- Taken together, they suggest the existence of multiple representational gradients in the sensory pathways of the human brain.

**Chapter 6** In Chapter 6, we used recurrent neural networks as a forward model for modeling the feature-response transformation. We also used several linear convolution models for the same purpose as the RNNs. We combined them with either a low-level feature space (spatial envelope (Oliva & Torralba, 2001)) or a high-level feature space (word embedding (Mikolov et al., 2013a)) to predict stimulus-evoked responses. We analyzed the same dataset as the one in Chapter 4 (natural movie-human visual cortex fMRI (Nishimoto et al., 2011a; Nishimoto et al., 2014)). Note that the approach in this chapter was different from the one in Chapters 2-5, where we exclusively used artificial neural networks (i.e., deep neural networks) as a feature space for modeling the stimulus-feature transformation.

We first compared the forward models and showed that RNNs have better encoding performance than the linear convolution models. We then cross-correlated the word embeddings and the spatial envelopes with the RNN hidden states. As a result, we showed that the majority of the RNN hidden units had optimal lags up to -20 s, reflecting hemodynamic factors. The remaining RNN hidden units had optimal lags beyond this period, reflecting other cognitive/neuronal factors or factors related to stimulus/feature statistics. The RNN hidden units that were combined with word embeddings had longer optimal lags than those that were coupled with spatial envelopes, reflecting differences in feature statistics. We finally stimulated the RNNs with an impulse and showed

that the impulse response of the RNNs is very similar to the hemodynamic response function.

The key outcomes of this chapter were the following:

- RNNs can learn not only the feature-response transformation (better than linear convolution models) but also the temporal dependencies of the responses.
- RNNs can estimate hemodynamic response functions in time-continuous fMRI experiments.
- Word embeddings can model neural representations of semantic contents of perceived natural movies.

**Overview** Table 7.1 gives an overview of the contemporary studies probing brain function with artificial neural networks, including Chapters 2-6.

**Outlook** In this thesis, we used artificial neural networks to probe human brain function. In the future, one may witness the development of increasingly sophisticated ANN architectures that explain more and more of the variance in the neural data acquired as subjects engage in cognitively demanding tasks.

A key advantage of the use of ANNs to probe human brain function is that it provides a computational model whose aim is to *explain* patterns of brain activity. For example, the main contribution of the use of DNNs to reveal a representational gradient in the visual ventral stream is that it provides an explanation of how such a gradient may arise (namely hierarchical processing of stimuli in service of object categorization).

It is of importance to be explicit on how far we can take the analogy between artificial neural networks and their biological

counterparts. Surely, the human brain is orders of magnitude more complex than the ANNs that have been developed to date and the employed artificial neurons ignore most of the intricacies of biological neurons. Still, the ambition to model cognitive processes using neural networks whose internal states can subsequently be used to predict neural response patterns is a promising endeavor. In this sense, reducing the gap between artificial and biological neural networks is a way to improve the sophistication of this approach.

An important related question is to what extent training of artificial neural networks reflects learning in biological neural networks. Supervised learning based on labeled data is unlikely to be a good model of biological learning. Unsupervised learning of the invariances that constitute our environment seems a safer bet. An even more promising approach is the implementation of reinforcement learning using artificial neural networks (Mnih et al., 2015). Theoretical arguments that reinforcement learning cannot be implemented in a biologically plausible manner using artificial neural networks are becoming obsolete due to new algorithmic developments (Lillicrap, Cownden, Tweed & Akerman, 2014; Rombouts, Bohte, Martinez-Trujillo & Roelfsema, 2015; Scellier & Bengio, 2017).

If we consider the architectures that have been used to date, one can expect the development and application of ever more sophisticated architectures. The use of recurrent neural networks has already been shown to provide a good account of neural dynamics. The combination of deep and recurrent architectures, as well as the use of other ways to endow ANNs with memory, such as the use of gated recurrent units (Cho et al., 2014) or neural Turing machines (Graves, 2013) will further advance this line of work.

Recent work also shows that RNNs can be used to model cognitive processing in several experimental tasks in a biologically plausible

manner (Song, Yang & Wang, 2016). We envision that such RNNs will start to be used to track the neural dynamics of high-level cognitive tasks that encompass the whole perception-action loop.

## 7.2 Conclusion

It has been more than 50 years since Hubel and Wiesel (1962) described their simple and complex cell model, and 40 years since Marr and Poggio (1976) put forth their Tri-Level Hypothesis. Since then, there has been ever more sophisticated attempts at understanding neural information processing, which came with their own fair share of debate as to what form an ideal model of neural computation should take and what level of explanation it should provide (Crick, 1989; Carandini, 2005).

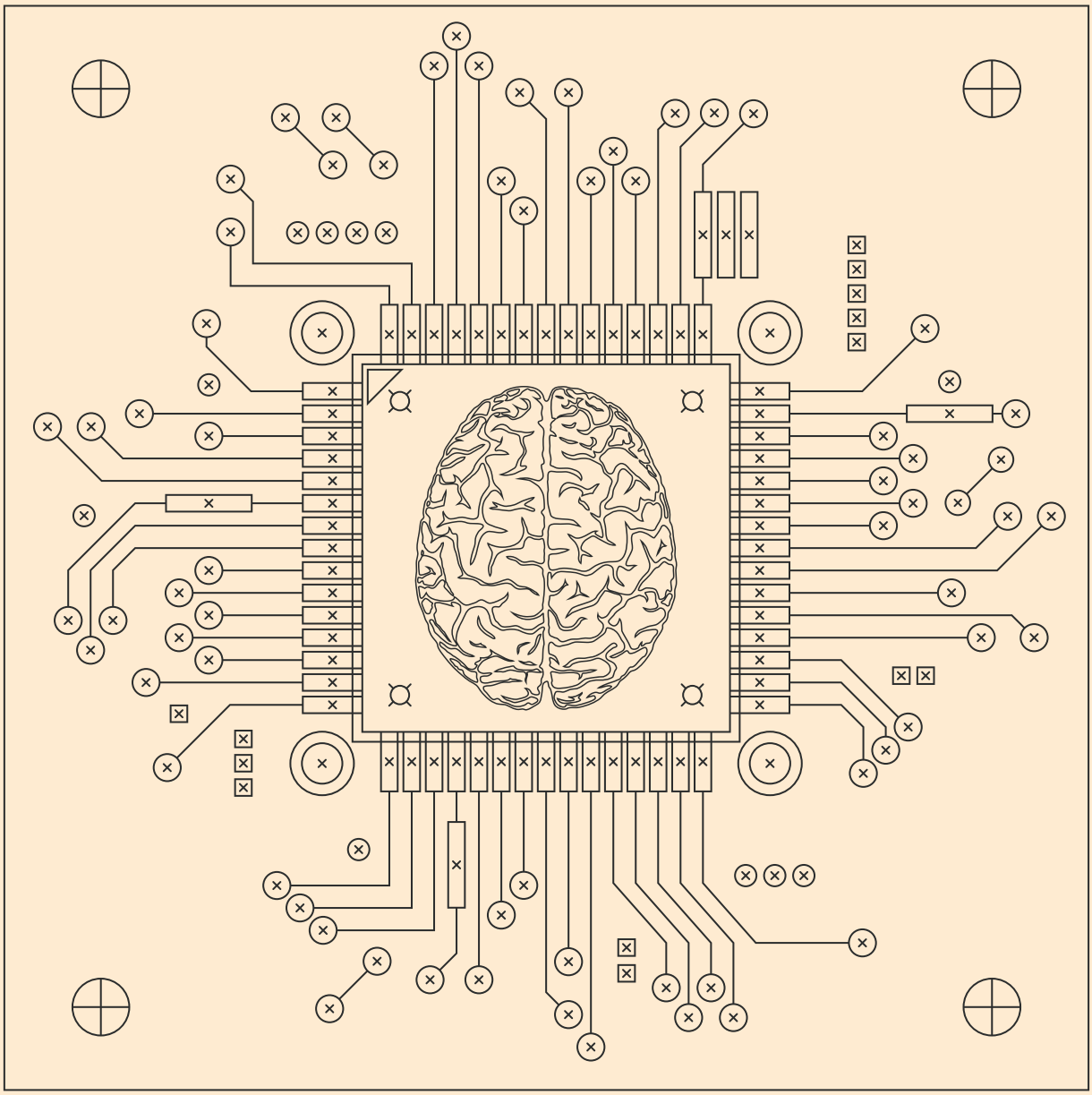
Throughout this thesis and the recent literature, it has been shown that artificial neural networks predict neural data better than their alternatives while explaining neural computation at an algorithmic level (e.g., task optimization). However, it does not follow from this fact that they also do so at an implementational level.

Considering the recent developments in biologically plausible learning rules, network architectures and objective functions, we expect artificial neural networks to be used for explaining not only the functions (e.g., working memory) but also the mechanisms (e.g., feedback projection) of neural computation.

Regardless of the aforementioned debates, we advocate testing alternative hypotheses about neural information processing by comparing generative models of neural data.

**Table 7.1:** Overview of contemporary studies probing brain function with artificial neural networks. ***ENC***: Encoding. ***DEC***: Decoding. ***RSA***: Representational similarity analysis. ***VEN***: Ventral stream. ***DOR***: Dorsal stream. ***AUD***: Auditory cortex. ***fMRI***: Functional magnetic resonance imaging. ***MEA***: Multielectrode array. ***MEG***: Magnetoencephalography. ***DNN***: Deep neural network (*ul*: unsupervised learning). ***MLP***: Multilayer perceptron (*ul*: unsupervised learning, *we*: word embedding). ***RNN***: Recurrent neural network. ***IMG***: Image. ***MOV***: Movie. ***MUS***: Music.

| | **Analysis** | | | **Area** | | | **Imaging** | | | **Model** | | | **Stimulus** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **ENC** | **DEC** | **RSA** | **VEN** | **DOR** | **AUD** | **fMRI** | **MEA** | **MEG** | **DNN** | **MLP** | **RNN** | **IMG** | **MOV** | **MUS** |
| van Gerven, de Lange and Heskes (2010) | | ✓ | | ✓ | | | ✓ | | | ✓(ul) | | | ✓ | | |
| Güçlü and van Gerven (2013) | ✓ | ✓ | | ✓ | | | ✓ | | | | ✓(ul) | | ✓ | | |
| Yamins et al. (2014) | ✓ | | ✓ | ✓ | | | | ✓ | | ✓ | | | ✓ | | |
| Agrawal, Stansbury, Malik and Gallant (2014) | ✓ | | | ✓ | | | ✓ | | | ✓ | | | ✓ | | |
| Chapter 2 | ✓ | ✓ | | ✓ | | | ✓ | | | | ✓(ul) | | ✓ | | |
| Khaligh-Razavi and Kriegeskorte (2014) | | | ✓ | ✓ | | | ✓ | ✓ | | ✓ | | | ✓ | | |
| Cadieu et al. (2014) | | ✓ | | ✓ | | | | ✓ | | ✓ | | | ✓ | | |
| Joukes, Hartmann and Krekelberg (2014) | | ✓ | | | ✓ | | | ✓ | | | | ✓ | | ✓ | |
| Chapter 3 | ✓ | ✓ | | ✓ | | | ✓ | | | ✓ | | | ✓ | | |
| Güçlü and van Gerven (2015) | ✓ | | | ✓ | | | ✓ | | | | ✓(we) | | ✓ | | |
| Chapter 4 | ✓ | ✓ | | | ✓ | | ✓ | | | ✓ | | | | ✓ | |
| Seibert et al. (2016) | | ✓ | ✓ | ✓ | | | ✓ | | | ✓ | | | ✓ | | |
| Cichy, Khosla, Pantazis, Torralba and Oliva (2016) | | | ✓ | sensor space | | | | | ✓ | ✓ | | | ✓ | | |
| Chapter 5 | | | ✓ | | | ✓ | ✓ | | | ✓ | | | | | ✓ |
| Chapter 6 | ✓ | | | | ✓ | | ✓ | | | | ✓(we) | ✓ | | ✓ | |
| Seeliger et al. (2017) | ✓ | ✓ | | ✓ | ✓ | | | | ✓ | ✓ | | | ✓ | | |
| Eickenberg, Gramfort, Varoquaux and Thirion (2017) | ✓ | | | ✓ | ✓ | | ✓ | | | ✓ | | | ✓ | ✓ | |
| Güçlütürk et al. (2017) | ✓ | ✓ | | ✓ | | | ✓ | | | ✓ | | | ✓ | | |
| Horikawa and Kamitani (2017) | | ✓ | | ✓ | | | ✓ | | | ✓ | | | ✓ | | |
| Cichy, Khosla, Pantazis and Oliva (2017) | | | ✓ | sensor space | | | | | ✓ | ✓ | | | ✓ | | |



# Bibliography


Ackley, D. H., Hinton, G. E. & Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines∗. *Cognitive Science*, *9*(1), 147–169. doi:10.1207/s15516709cog0901_7. (Cit. on p. 16)

Agrawal, P., Stansbury, D., Malik, J. & Gallant, J. L. (2014). Pixels to voxels: Modeling visual representation in the human brain. *CoRR*, *abs/1407.5104*. (Cit. on pp. 11, 19, 78, 103, 113, 130, 174).

Alluri, V., Toiviainen, P., Lund, T. E., Wallentin, M., Vuust, P., Nandi, A. K., ... Brattico, E. (2013). From vivaldi to beatles and back: Predicting lateralized brain responses to music. *NeuroImage*, *83*, 627–636. doi:10.1016/j.neuroimage.2013.06.064. (Cit. on p. 112)

Alluri, V., Toiviainen, P., Jääskeläinen, I. P., Glerean, E., Sams, M. & Brattico, E. (2012). Large-scale brain networks emerge from dynamic processing of musical timbre, key and rhythm. *NeuroImage*, *59*(4), 3677–3689. doi:10.1016/j.neuroimage.2011.11.019. (Cit. on p. 112)

Aquino, K., Robinson, P. & Drysdale, P. (2014). Spatiotemporal hemodynamic response functions derived from physiology. *Journal of Theoretical Biology*, *347*, 118–136. doi:10.1016/j.jtbi.2013.12.027. (Cit. on pp. 80, 153)

Barlow, H. B. (2012). Possible principles underlying the transformations of sensory messages. In *Sensory communication* (pp. 216–234). The MIT Press. doi:10.7551/mitpress/9780262518420.003.0013. (Cit. on p. 27)

Bell, A. J. & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision Research*, *37*(23), 3327–3338. doi:10.1016/s0042-6989(97)00121-1. (Cit. on p. 27)

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends® in Machine Learning*, *2*(1), 1–127. doi:10.1561/2200000006. (Cit. on p. 11)

Bengio, Y., Simard, P. & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157–166. doi:10.1109/72.279181. (Cit. on p. 18)

Bengio, Y., Courville, A. & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828. doi:10.1109/tpami.2013.50. (Cit. on p. 52)

Bengio, Y., Ducharme, R., Vincent, P. & Janvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, *3*, 1137–1155. (Cit. on p. 14).

Blasdel, G. G. (1992). Orientation selectivity, preference, and continuity in monkey striate cortex. *Journal of Neuroscience*, *12*(8), 3139–3161. (Cit. on p. 41).

Boyd, S. & Vandenberghe, L. (2004). *Convex optimization* (1st ed.). Cambridge University Press. (Cit. on p. 34).

Brown, E. N., Kass, R. E. & Mitra, P. P. (2004). Multiple neural spike train data analysis: State-of-the-art and future challenges. *Nature Neuroscience*, *7*(5), 456–461. doi:10.1038/nn1228. (Cit. on p. 26)

Cadieu, C. F., Hong, H., Yamins, D. L. K., Pinto, N., Ardila, D., Solomon, E. A., ... DiCarlo, J. J. (2014). Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Computational Biology*, *10*(12), e1003963. doi:10.1371/journal.pcbi.1003963. (Cit. on pp. 12, 80, 103, 113, 130, 174)

Carandini, M. (2005). Do we know what the early visual system does? *Journal of Neuroscience*, *25*(46), 10577–10597. doi:10.1523/jneurosci.3726-05.2005. (Cit. on pp. 81, 173)

Casey, M., Thompson, J., Kang, O., Raizada, R. & Wheatley, T. (2012). Population codes representing musical timbre for high-level fMRI categorization of music genres. In *Lecture notes in computer science* (pp. 34–41). Springer Berlin Heidelberg. doi:10.1007/978-3-642-34713-9_5. (Cit. on pp. 112, 115)

Chatfield, K., Simonyan, K., Vedaldi, A. & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. *CoRR*, *abs/1405.3531*. (Cit. on pp. 59, 62, 166).

Cho, K., van Merrienboer, B., Gülçehre, Ç., Bougares, F., Schwenk, H. & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, *abs/1406.1078*. (Cit. on pp. 131, 141, 172).

Chollet, F. (2015). Keras. https://github.com/fchollet/keras. (Cit. on p. 117).

Cichy, R. M., Khosla, A., Pantazis, D., Torralba, A. & Oliva, A. (2016). Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. *Scientific Reports*, *6*(1). doi:10.1038/srep27755. (Cit. on pp. 12, 113, 130, 137, 174)

Cichy, R. M., Khosla, A., Pantazis, D. & Oliva, A. (2017). Dynamics of scene representations in the human brain revealed by magnetoencephalography and deep neural networks. *NeuroImage*, *153*, 346–358. doi:10.1016/j.neuroimage.2016.03.063. (Cit. on pp. 113, 174)

Copeland, B. J. & Proudfoot, D. (1996). On alan turing's anticipation of connectionism. *Synthese*, *108*(3), 361–377. doi:10.1007/bf00413694. (Cit. on p. 5)

Cox, D. D. (2014). Do we understand high-level vision? *Current Opinion in Neurobiology*, *25*, 187–193. doi:10.1016/j.conb.2014.01.016. (Cit. on p. 56)

Creutzfeldt, O. D. & Nothdurft, H. C. (1978). Representation of complex visual stimuli in the brain. *Naturwissenschaften*, *65*(6), 307–318. doi:10.1007/bf00368371. (Cit. on p. 3)

Crick, F. (1989). The recent excitement about neural networks. *Nature*, *337*(6203), 129–132. doi:10.1038/337129a0. (Cit. on p. 173)

Çukur, T., Nishimoto, S., Huth, A. G. & Gallant, J. L. (2013). Attention during natural vision warps semantic representation across the human brain. *Nature Neuroscience*, *16*(6), 763–770. doi:10.1038/nn.3381. (Cit. on pp. 51, 78)

Cybenko, G. (1992). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, *5*(4), 455–455. doi:10.1007/bf02134016. (Cit. on p. 9)

Dale, A. M. (1999). Optimal experimental design for event-related fMRI. *Human Brain Mapping*, *8*(2-3), 109–114. doi:10.1002/(sici)1097-0193(1999)8:2/3<109::aid-hbm7>3.0.co;2-w. (Cit. on p. 153)

Daugman, J. G. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A*, *2*(7), 1160. doi:10.1364/josaa.2.001160. (Cit. on p. 40)

Dayan, P. & Abbott, L. F. (2005). *Theoretical neuroscience: Computational and mathematical modeling of neural systems (computational neuroscience series)* (Revised ed.). The MIT Press. (Cit. on p. 26).

DeAngelis, G. C., Ghose, G. M., Ohzawa, I. & Freeman, R. D. (1999). Functional micro-organization of primary visual cortex: Receptive field analysis of nearby neurons. *Journal of Neuroscience*, *19*(10), 4046–4064. (Cit. on p. 41).

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. IEEE. doi:10.1109/cvpr.2009.5206848. (Cit. on p. 60)

Desimone, R., Albright, T. D., Gross, C. G. & Bruce, C. (1984). Stimulus-selective properties of inferior temporal neurons in the macaque. *Journal of Neuroscience*, *4*(8), 2051–2062. (Cit. on p. 77).

DiCarlo, J. J. & Cox, D. D. (2007). Untangling invariant object recognition. *Trends in Cognitive Sciences*, *11*(8), 333–341. doi:10.1016/j.tics.2007.06.010. (Cit. on p. 76)

DiCarlo, J. J., Zoccolan, D. & Rust, N. C. (2012). How does the brain solve visual object recognition? *Neuron*, *73*(3), 415–434. doi:10.1016/j.neuron.2012.01.010. (Cit. on pp. 150, 155)

Dieleman, S. & Schrauwen, B. (2014). End-to-end learning for music audio. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. doi:10.1109/icassp.2014.6854950. (Cit. on pp. 104, 120)

Dieleman, S. & Schrauwen, B. (2013). Multiscale approaches to music audio feature learning. In A. de Souza Britto Jr., F. Gouyon & S. Dixon (Eds.), *Proceedings of the 14th international society for music information retrieval conference, ISMIR 2013, curitiba, brazil, november 4-8, 2013* (pp. 3–8). (Cit. on p. 120).

Dosovitskiy, A. & Brox, T. (2015). Inverting convolutional networks with convolutional networks. *CoRR*, *abs/1506.02753*. (Cit. on p. 103).

Dumoulin, S. O. & Wandell, B. A. (2008). Population receptive field estimates in human visual cortex. *NeuroImage*, *39*(2), 647–660. doi:10.1016/j.neuroimage.2007.09.034. (Cit. on pp. 43, 57)

Duyn, J. H. (2012). The future of ultra-high field MRI and fMRI for study of the human brain. *NeuroImage*, *62*(2), 1241–1248. doi:10.1016/j.neuroimage.2011.10.065. (Cit. on p. 53)

Edelman, A., Arias, T. A. & Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, *20*(2), 303–353. doi:10.1137/s0895479895290954. (Cit. on p. 34)

Eickenberg, M. (2015). *Evaluating computational models of vision with functional magnetic resonance imaging* (Doctoral dissertation, University of Paris-Sud). (Cit. on p. 103).

Eickenberg, M., Gramfort, A., Varoquaux, G. & Thirion, B. (2017). Seeing it all: Convolutional network layers map the function of the human visual system. *NeuroImage*, *152*, 184–194. doi:10.1016/j.neuroimage.2016.10.001. (Cit. on pp. 12, 19, 130, 174)

Elman, J. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. doi:10.1016/0364-0213(90)90002-e. (Cit. on p. 17)

Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition*, *48*(1), 71–99. doi:10.1016/0010-0277(93)90058-4. (Cit. on p. 160)

Emadi, N. & Esteky, H. (2014). Behavioral demand modulates object category representation in the inferior temporal cortex. *Journal of Neurophysiology*, *112*(10), 2628–2637. doi:10.1152/jn.00761.2013. (Cit. on p. 78)

Fei-Fei, L., Fergus, R. & Perona, P. (2007). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, *106*(1), 59–70. doi:10.1016/j.cviu.2005.09.012. (Cit. on pp. 38, 45, 165)

Felsen, G. & Dan, Y. (2005). A natural approach to studying vision. *Nature Neuroscience*, *8*(12), 1643–1646. doi:10.1038/nn1608. (Cit. on pp. 3, 77, 130)

Forstmann, B. U. & Wagenmakers, E.-J. (Eds.). (2015). *An introduction to model-based cognitive neuroscience*. Springer New York. doi:10.1007/978-1-4939-2236-9. (Cit. on p. 81)

Friston, K. J., Holmes, A. P., Worsley, K. J., Poline, J.-P., Frith, C. D. & Frackowiak, R. S. J. (1994). Statistical parametric maps in functional imaging: A general linear approach. *Human Brain Mapping*, *2*(4), 189–210. doi:10.1002/hbm.460020402. (Cit. on p. 137)

Friston, K. J., Josephs, O., Rees, G. & Turner, R. (1998). Nonlinear event-related responses in fMRI. *Magnetic Resonance in Medicine*, *39*(1), 41–52. doi:10.1002/mrm.1910390109. (Cit. on pp. 138, 153)

Friston, K., Harrison, L. & Penny, W. (2003). Dynamic causal modelling. *NeuroImage*, *19*(4), 1273–1302. doi:10.1016/s1053-8119(03)00202-7. (Cit. on p. 159)

Friston, K., Mechelli, A., Turner, R. & Price, C. (2000). Nonlinear responses in fMRI: The balloon model, volterra kernels, and other hemodynamics. *NeuroImage*, *12*(4), 466–477. doi:10.1006/nimg.2000.0630. (Cit. on p. 132)

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202. doi:10.1007/bf00344251. (Cit. on pp. 2, 10, 80, 85)

Furmanski, C. S. & Engel, S. A. (2000). *Nature Neuroscience*, *3*(6), 535–536. doi:10.1038/75702. (Cit. on p. 43)

Fuster, J. M. (2003). *Cortex and mind: Unifying cognition* (1st ed.). Oxford University Press. (Cit. on p. 112).

Fyshe, A., Murphy, B., Talukdar, P. P. & Mitchell, T. M. (2013). Documents and dependencies: An exploration of vector space models for semantic composition. In J. Hockenmaier & S. Riedel (Eds.), *Proceedings of the seventeenth conference on computational natural language learning, conll 2013, sofia, bulgaria, august 8-9, 2013* (pp. 84–93). ACL. (Cit. on pp. 15, 130).

Glorot, X. & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Y. W. Teh & D. M. Titterington (Eds.), *Proceedings of the thirteenth international conference on artificial intelligence and statistics, AISTATS 2010, chia laguna resort, sardinia, italy, may 13-15, 2010* (Vol. 9, pp. 249–256). JMLR Proceedings. JMLR.org. (Cit. on p. 116).

Glover, G. H. (1999). Deconvolution of impulse response in event-related BOLD fMRI1. *NeuroImage*, *9*(4), 416–429. doi:10.1006/nimg.1998.0419. (Cit. on p. 153)

Goodale, M. A. & Milner, A. (1992). Separate visual pathways for perception and action. *Trends in Neurosciences*, *15*(1), 20–25. doi:10.1016/0166-2236(92)90344-8. (Cit. on pp. 78, 84)

Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke, H. & Schmidhuber, J. (2009). A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(5), 855–868. doi:10.1109/tpami.2008.137. (Cit. on p. 131)

Graves, A. (2013). Generating sequences with recurrent neural networks. *CoRR*, *abs/1308.0850*. (Cit. on pp. 131, 172).

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R. & Schmidhuber, J. (2015). LSTM: A search space odyssey. *CoRR*, *abs/1503.04069*. (Cit. on p. 103).

Gregor, K., Danihelka, I., Graves, A. & Wierstra, D. (2015). DRAW: A recurrent neural network for image generation. *CoRR*, *abs/1502.04623*. (Cit. on p. 156).

Grill-Spector, K. & Weiner, K. S. (2014). The functional architecture of the ventral temporal cortex and its role in categorization. *Nature Reviews Neuroscience*, *15*(8), 536–548. doi:10.1038/nrn3747. (Cit. on p. 77)

Grill-Spector, K., Kushnir, T., Hendler, T., Edelman, S., Itzchak, Y. & Malach, R. (1998). A sequence of object-processing stages revealed by fMRI in the human occipital lobe. *Human Brain Mapping*, *6*(4), 316–328. doi:10.1002/(sici)1097-0193(1998)6:4<316::aid-hbm9>3.0.co;2-6. (Cit. on p. 76)

Groen, I. I. A., Ghebreab, S., Prins, H., Lamme, V. A. F. & Scholte, H. S. (2013). From image statistics to scene gist: Evoked neural activity reveals transition from low-level natural image structure to scene category. *Journal of Neuroscience*, *33*(48), 18814–18824. doi:10.1523/jneurosci.3128-13.2013. (Cit. on p. 137)

Gross, C. G., Rocha-Miranda, C. E. & Bender, D. B. (1972). Visual properties of neurons in inferotemporal cortex of the macaque. *Journal of Neurophysiology*, *35*(1), 96–111. (Cit. on p. 56).

Gutmann, M. & Hyvärinen, A. (2012). Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, *13*, 307–361. (Cit. on p. 52).

Gutmann, M. U. & Hyvärinen, A. (2013). A three-layer model of natural image statistics. *Journal of Physiology-Paris*, *107*(5), 369–398. doi:10.1016/j.jphysparis.2013.01.001. (Cit. on p. 52)

Güçlü, U. & van Gerven, M. A. J. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, *35*(27), 10005–10014. doi:10.1523/jneurosci.5023-14.2015. (Cit. on pp. 84, 96, 102–104, 113, 124, 126, 130, 132, 150, 155)

Güçlü, U. & van Gerven, M. A. J. (2015). Semantic vector space models predict neural responses to complex visual stimuli. *ArXiv e-prints*. arXiv: 1510.04738 [q-bio.NC]. (Cit. on pp. 15, 19, 130, 174)

Güçlü, U. & van Gerven, M. (2013). Unsupervised learning of features for bayesian decoding in functional magnetic resonance imaging. In *Belgian-dutch conference on machine learning*. (Cit. on pp. 16, 19, 174).

Güçlü, U. & van Gerven, M. A. J. (2014). Unsupervised feature learning improves prediction of human brain activity in response to natural images. *PLoS Computational Biology*, *10*(8), e1003724. doi:10.1371/journal.pcbi.1003724. (Cit. on pp. 56, 60, 68, 92, 96, 139)

Güçlü, U. & van Gerven, M. A. (2017). Increasingly complex representations of natural movies across the dorsal stream are shared between subjects. *NeuroImage*, *145*, 329–336. doi:10.1016/j.neuroimage.2015.12.036. (Cit. on pp. 113, 124, 130, 132)

Güçlütürk, Y., Güçlü, U., Seeliger, K., Bosch, S., van Lier, R. & van Gerven, M. (2017). Deep adversarial neural decoding. *CoRR*, *abs/1705.07109*. (Cit. on pp. 13, 174).

Hanke, M., Dinga, R., Häusler, C., Guntupalli, J. S., Casey, M., Kaule, F. R. & Stadler, J. (2015). High-resolution 7-tesla fMRI data on the perception of musical genres – an extension to the studyforrest dataset. *F1000Research*. doi:10.12688/f1000research.6679.1. (Cit. on pp. 115, 119, 169)

Hanke, M., Halchenko, Y. O., Sederberg, P. B., Hanson, S. J., Haxby, J. V. & Pollmann, S. (2009). PyMVPA: A python toolbox for multivariate pattern analysis of fMRI data. *Neuroinformatics*, *7*(1), 37–53. doi:10.1007/s12021-008-9041-y. (Cit. on p. 87)

Hanley, J. A. & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, *143*(1), 29–36. doi:10.1148/radiology.143.1.7063747. (Cit. on p. 102)

Hansen, K. A., David, S. V. & Gallant, J. L. (2004). Parametric reverse correlation reveals spatial linearity of retinotopic human v1 BOLD response. *NeuroImage*, *23*(1), 233–241. doi:10.1016/j.neuroimage.2004.05.012. (Cit. on pp. 87, 134)

Harrison, S. A. & Tong, F. (2009). Decoding reveals the contents of visual working memory in early visual areas. *Nature*, *458*(7238), 632–635. doi:10.1038/nature07832. (Cit. on p. 78)

Hastie, T., Tibshirani, R. & Friedman, J. (2009). *The elements of statistical learning*. Springer New York. doi:10.1007/978-0-387-84858-7. (Cit. on p. 35)

Haxby, J. V. (2001). Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, *293*(5539), 2425–2430. doi:10.1126/science.1063736. (Cit. on pp. 13, 27)

Haxby, J. V., Grady, C. L., Horwitz, B., Ungerleider, L. G., Mishkin, M., Carson, R. E., ... Rapoport, S. I. (1991). Dissociation of object and spatial visual processing pathways in human extrastriate cortex. *Proceedings of the National Academy of Sciences*, *88*(5), 1621–1625. doi:10.1073/pnas.88.5.1621. (Cit. on p. 84)

Haxby, J. V., Guntupalli, J. S., Connolly, A. C., Halchenko, Y. O., Conroy, B. R., Gobbini, M. I., ... Ramadge, P. J. (2011). A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron*, *72*(2), 404–416. doi:10.1016/j.neuron.2011.08.026. (Cit. on pp. 65, 86, 87, 105, 168)

Haykin, S. (1994). *Neural networks: A comprehensive foundation*. Macmillan Coll Div. (Cit. on p. 5).

He, K., Zhang, X., Ren, S. & Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, *abs/1512.03385*. (Cit. on p. 10).

Hebb, D. (2002). *The organization of behavior: A neuropsychological theory* (1st ed.). Psychology Press. (Cit. on p. 2).

Hegdé, J. & Essen, D. C. V. (2006). A comparative study of shape representation in macaque visual areas v2 and v4. *Cerebral Cortex*, *17*(5), 1100–1116. doi:10.1093/cercor/bhl020. (Cit. on p. 77)

Hinton, G. E. (2007). Learning multiple layers of representation. *Trends in Cognitive Sciences*, *11*(10), 428–434. doi:10.1016/j.tics.2007.09.004. (Cit. on p. 79)

Hinton, G. E., Osindero, S. & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–1554. doi:10.1162/neco.2006.18.7.1527. (Cit. on pp. 16, 53)

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, *abs/1207.0580*. (Cit. on pp. 60, 90, 142, 156).

Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735. (Cit. on pp. 18, 103, 131, 139)

Hochstein, S. & Ahissar, M. (2002). View from the top. *Neuron*, *36*(5), 791–804. doi:10.1016/s0896-6273(02)01091-7. (Cit. on p. 80)

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, *79*(8), 2554–2558. doi:10.1073/pnas.79.8.2554. (Cit. on p. 16)

Horikawa, T., Tamaki, M., Miyawaki, Y. & Kamitani, Y. (2013). Neural decoding of visual imagery during sleep. *Science*, *340*(6132), 639–642. doi:10.1126/science.1234330. (Cit. on p. 78)

Horikawa, T. & Kamitani, Y. (2015). Generic decoding of seen and imagined objects using hierarchical visual features. *CoRR*, *abs/1510.06479*. (Cit. on p. 113).

Horikawa, T. & Kamitani, Y. (2017). Generic decoding of seen and imagined objects using hierarchical visual features. *Nature Communications*, *8*, 15037. doi:10.1038/ncomms15037. (Cit. on pp. 13, 174)

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251–257. doi:10.1016/0893-6080(91)90009-t. (Cit. on p. 9)

Hubel, D. H. & Wiesel, T. N. (1977). Ferrier lecture: Functional architecture of macaque monkey visual cortex. *Proceedings of the Royal Society B: Biological Sciences*, *198*(1130), 1–59. doi:10.1098/rspb.1977.0085. (Cit. on p. 41)

Hubel, D. H. & Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of Physiology*, *195*(1), 215–243. doi:10.1113/jphysiol.1968.sp008455. (Cit. on p. 39)

Hubel, D. H. & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, *160*(1), 106–154. doi:10.1113/jphysiol.1962.sp006837. (Cit. on pp. 56, 173)

Hung, C. P. (2005). Fast readout of object identity from macaque inferior temporal cortex. *Science*, *310*(5749), 863–866. doi:10.1126/science.1117593. (Cit. on p. 56)

Huth, A. G., Nishimoto, S., Vu, A. T. & Gallant, J. L. (2012). A continuous semantic space describes the representation of thousands of object and action categories across the human brain. *Neuron*, *76*(6), 1210–1224. doi:10.1016/j.neuron.2012.10.014. (Cit. on pp. 15, 76, 130)

Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, *6*, 695–709. (Cit. on p. 52).

Hyvärinen, A. & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks*, *13*(4-5), 411–430. doi:10.1016/s0893-6080(00)00026-5. (Cit. on p. 16)

Hyvärinen, A. (2010). Statistical models of natural images and cortical visual representation. *Topics in Cognitive Science*, *2*(2), 251–264. doi:10.1111/j.1756-8765.2009.01057.x. (Cit. on pp. 28, 51)

Hyvärinen, A. & Hoyer, P. O. (2001). A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, *41*(18), 2413–2423. doi:10.1016/s0042-6989(01)00114-6. (Cit. on pp. 31, 38, 41, 51, 164)

Ioffe, S. & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, *abs/1502.03167*. (Cit. on p. 117).

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R. B., ... Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *CoRR*, *abs/1408.5093*. (Cit. on pp. 60, 62, 90).

Jones, J. & Palmer, L. A. (1987). An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, *58*(6), 1233–1258. (Cit. on pp. 39, 40, 56).

Jordan, M. I. (1997). Serial order: A parallel distributed processing approach. In *Neural-network models of cognition - biobehavioral foundations* (pp. 471–495). Elsevier. doi:10.1016/s0166-4115(97)80111-2. (Cit. on p. 17)

Joukes, J., Hartmann, T. S. & Krekelberg, B. (2014). Motion detection based on recurrent network dynamics. *Frontiers in Systems Neuroscience*, *8*. doi:10.3389/fnsys.2014.00239. (Cit. on pp. 18, 19, 174)

Kamitani, Y. & Tong, F. (2005). Decoding the visual and subjective contents of the human brain. *Nature Neuroscience*, *8*(5), 679–685. doi:10.1038/nn1444. (Cit. on pp. 13, 27)

Kanitscheider, I. & Fiete, I. (2016). Training recurrent networks to generate hypotheses about how the brain solves hard navigation problems. *ArXiv e-prints*. arXiv: 1609.09059 [q-bio.NC]. (Cit. on p. 156)

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R. & Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *2014 IEEE conference on computer vision and pattern recognition*. IEEE. doi:10.1109/cvpr.2014.223. (Cit. on p. 89)

Karpathy, A., Johnson, J. & Li, F. (2015). Visualizing and understanding recurrent networks. *CoRR*, *abs/1506.02078*. (Cit. on p. 158).

Kay, K. N., Winawer, J., Mezer, A. & Wandell, B. A. (2013a). Compressive spatial summation in human visual cortex. *Journal of Neurophysiology*, *110*(2), 481–494. doi:10.1152/jn.00105.2013. (Cit. on pp. 49, 95, 143)

Kay, K. N., Winawer, J., Rokem, A., Mezer, A. & Wandell, B. A. (2013b). A two-stage cascade model of BOLD responses in human visual cortex. *PLoS Computational Biology*, *9*(5), e1003079. doi:10.1371/journal.pcbi.1003079. (Cit. on pp. 27, 40, 53)

Kay, K. N., Rokem, A., Winawer, J., Dougherty, R. F. & Wandell, B. A. (2013c). GLMdenoise: A fast, automated technique for denoising task-based fMRI data. *Frontiers in Neuroscience*, *7*. doi:10.3389/fnins.2013.00247. (Cit. on p. 116)

Kay, K. N., Naselaris, T., Prenger, R. J. & Gallant, J. L. (2008). Identifying natural images from human brain activity. *Nature*, *452*(7185), 352–355. doi:10.1038/nature06713. (Cit. on pp. 9, 13, 27, 28, 30, 37, 40, 43, 45, 56, 57, 62, 68, 96, 164, 166)

Kell, A., Yamins, D., Norman-Haignere, S. & McDermott, J. (2016). Speech-trained neural networks behave like human listeners and reveal a hierarchy in auditory cortex. In *Computational and systems neuroscience*. (Cit. on pp. 13, 113, 124).

Khaligh-Razavi, S.-M. & Kriegeskorte, N. (2014). Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Computational Biology*, *10*(11), e1003915. doi:10.1371/journal.pcbi.1003915. (Cit. on pp. 11, 12, 17, 68, 80, 103, 113, 130, 174)

Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, *abs/1412.6980*. (Cit. on pp. 116, 141).

Knowles, D. & Ghahramani, Z. (2011). Nonparametric bayesian sparse factor models with application to gene expression modeling. *The Annals of Applied Statistics*, *5*(2B), 1534–1552. doi:10.1214/10-aoas435. (Cit. on p. 51)

Kok, P., Brouwer, G. J., van Gerven, M. A. J. & de Lange, F. P. (2013). Prior expectations bias sensory representations in visual cortex. *Journal of Neuroscience*, *33*(41), 16275–16284. doi:10.1523/jneurosci.0742-13.2013. (Cit. on p. 51)

Kok, P. & de Lange, F. P. (2014). Shape perception simultaneously up- and downregulates neural activity in the primary visual cortex. *Current Biology*, *24*(13), 1531–1535. doi:10.1016/j.cub.2014.05.042. (Cit. on p. 78)

Kriegeskorte, N., Goebel, R. & Bandettini, P. (2006). Information-based functional brain mapping. *Proceedings of the National Academy of Sciences*, *103*(10), 3863–3868. doi:10.1073/pnas.0600244103. (Cit. on p. 122)

Kriegeskorte, N. (2015). Deep neural networks: A new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science*, *1*(1), 417–446. doi:10.1146/annurev-vision-082114-035447. (Cit. on pp. 3, 132, 156)

Kriegeskorte, N. (2008). Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*. doi:10.3389/neuro.06.004.2008. (Cit. on pp. 11, 118, 122, 151, 169)

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou & K. Q. Weinberger (Eds.), *Advances in neural information processing systems 25: 26th annual conference on neural information processing systems 2012. proceedings of a meeting held december 3-6, 2012, lake tahoe, nevada, united states.* (pp. 1106–1114). (Cit. on pp. 11, 12, 59, 62, 88, 117, 169).

Köster, U. & Hyvärinen, A. (2010). A two-layer model of natural stimuli estimated with score matching. *Neural Computation*, *22*(9), 2308–2333. doi:10.1162/neco_a_00010. (Cit. on p. 52)

Larsson, J. & Heeger, D. J. (2006). Two retinotopic visual areas in human lateral occipital cortex. *Journal of Neuroscience*, *26*(51), 13128–13142. doi:10.1523/jneurosci.1657-06.2006. (Cit. on p. 78)

Lau, B., Stanley, G. B. & Dan, Y. (2002). Computational subunits of visual cortical neurons revealed by artificial neural networks. *Proceedings of the National Academy of Sciences*, *99*(13), 8974–8979. doi:10.1073/pnas.122173799. (Cit. on pp. 9, 19)

Law, E., West, K., Mandel, M. I., Bay, M. & Downie, J. S. (2009). Evaluation of algorithms using games: The case of music tagging. In K. Hirata, G. Tzanetakis & K. Yoshii (Eds.), *Proceedings of the 10th international society for music information retrieval conference, ISMIR 2009, kobe international conference center, kobe, japan, october 26-30, 2009* (pp. 387–392). International Society for Music Information Retrieval. (Cit. on pp. 114, 168).

Le, Q. V. (2013). Building high-level features using large scale unsupervised learning. In *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE. doi:10.1109/icassp.2013.6639343. (Cit. on p. 52)

Le, Q. V., Zou, W. Y., Yeung, S. Y. & Ng, A. Y. (2011). Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR 2011*. IEEE. doi:10.1109/cvpr.2011.5995496. (Cit. on p. 52)

LeCun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. doi:10.1038/nature14539. (Cit. on pp. 6, 10, 84, 88)

Lee, H., Grosse, R., Ranganath, R. & Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th annual international conference on machine learning - ICML '09*. ACM Press. doi:10.1145/1553374.1553453. (Cit. on p. 52)

Lee, H., Ekanadham, C. & Ng, A. Y. (2007). Sparse deep belief net model for visual area V2. In J. C. Platt, D. Koller, Y. Singer & S. T. Roweis (Eds.), *Advances in neural information processing systems 20, proceedings of the twenty-first annual conference on neural information processing systems, vancouver, british columbia, canada, december 3-6, 2007* (pp. 873–880). Curran Associates, Inc. (Cit. on p. 52).

Lee, T. S. (1996). Image representation using 2d gabor wavelets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *18*(10), 959–971. doi:10.1109/34.541406. (Cit. on p. 40)

Leeds, D. D., Seibert, D. A., Pyles, J. A. & Tarr, M. J. (2013). Comparing visual representations across human fMRI and computational vision. *Journal of Vision*, *13*(13), 25–25. doi:10.1167/13.13.25. (Cit. on p. 137)

Lehky, S. R., Sejnowski, T. J. & Desimone, R. (1992). Predicting responses of nonlinear neurons in monkey striate cortex to complex patterns. *Journal of Neuroscience*, *12*(9), 3568–3581. (Cit. on pp. 9, 19).

Lescroart, M., Kay, K., Naselaris, T., Prenger, R., Oliver, M. & Gallant, J. (2011). Fmri of human visual areas in response to natural images. CRCNS.org. doi:10.6080/k0qn64ng. (Cit. on pp. 28, 165, 166)

Lillicrap, T. P., Cownden, D., Tweed, D. B. & Akerman, C. J. (2014). Random feedback weights support learning in deep neural networks. *CoRR*, *abs/1411.0247*. (Cit. on p. 172).

Logothetis, N. K. & Wandell, B. A. (2004). Interpreting the BOLD signal. *Annual Review of Physiology*, *66*(1), 735–769. doi:10.1146/annurev.physiol.66.082602.092845. (Cit. on pp. 79, 131)

Mahendran, A. & Vedaldi, A. (2014). Understanding deep image representations by inverting them. *CoRR*, *abs/1412.0035*. (Cit. on p. 103).

Mansfield, R. J. W. (1974). Neural basis of orientation perception in primate vision. *Science*, *186*(4169), 1133–1135. doi:10.1126/science.186.4169.1133. (Cit. on p. 43)

Marĉelja, S. (1980). Mathematical description of the responses of simple cortical cells∗. *Journal of the Optical Society of America*, *70*(11), 1297. doi:10.1364/josa.70.001297. (Cit. on pp. 9, 12)

Markov, N. T., Vezoli, J., Chameau, P., Falchier, A., Quilodran, R., Huissoud, C., ... Kennedy, H. (2013). Anatomy of hierarchy: Feedforward and feedback pathways in macaque visual cortex. *Journal of Comparative Neurology*, *522*(1), 225–259. doi:10.1002/cne.23458. (Cit. on p. 69)

Marr, D. & Poggio, T. (1976). *From understanding computation to understanding neural circuitry*. Massachusetts Institute of Technology. (Cit. on p. 173).

McClelland, J. L. & Rumelhart, D. E. (1989). *Explorations in parallel distributed processing - macintosh version: A handbook of models, programs, and exercises*. A Bradford Book. (Cit. on p. 2).

McCulloch, W. S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115–133. doi:10.1007/bf02478259. (Cit. on p. 5)

McFarland, J. M., Cui, Y. & Butts, D. A. (2013). Inferring nonlinear neuronal computation based on physiologically plausible inputs. *PLoS Computational Biology*, *9*(7), e1003143. doi:10.1371/journal.pcbi.1003143. (Cit. on p. 117)

McKee, J. L., Riesenhuber, M., Miller, E. K. & Freedman, D. J. (2014). Task dependence of visual and category representations in prefrontal and inferior temporal cortices. *Journal of Neuroscience*, *34*(48), 16065–16075. doi:10.1523/jneurosci.1660-14.2014. (Cit. on p. 78)

Meeds, E. & Roweis, S. (2007). *Nonparametric bayesian biclustering*. University of Toronto. (Cit. on p. 65).

Mesulam, M. (1998). From sensation to cognition. *Brain*, *121*(6), 1013–1052. doi:10.1093/brain/121.6.1013. (Cit. on p. 78)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. & Dean, J. (2013a). Distributed representations of words and phrases and their compositionality. *CoRR*, *abs/1310.4546*. (Cit. on pp. 15, 136, 170).

Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013b). Efficient estimation of word representations in vector space. *CoRR*, *abs/1301.3781*. (Cit. on pp. 14, 136).

Mikolov, T., Yih, W. & Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In L. Vanderwende, H. D. III & K. Kirchhoff (Eds.), *Human language technologies: Conference of the north american chapter of the association of computational linguistics, proceedings, june 9-14, 2013, westin peachtree plaza hotel, atlanta, georgia, USA* (pp. 746–751). The Association for Computational Linguistics. (Cit. on p. 15).

Mishkin, M., Ungerleider, L. G. & Macko, K. A. (1983). Object vision and spatial vision: Two cortical pathways. *Trends in Neurosciences*, *6*, 414–417. doi:10.1016/0166-2236(83)90190-x. (Cit. on pp. 84, 150, 155)

Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K.-M., Malave, V. L., Mason, R. A. & Just, M. A. (2008). Predicting human brain activity associated with the meanings of nouns. *Science*, *320*(5880), 1191–1195. doi:10.1126/science.1152876. (Cit. on pp. 13, 15, 16, 27, 130)

Miyawaki, Y., Uchida, H., Yamashita, O., aki Sato, M., Morito, Y., Tanabe, H. C., ... Kamitani, Y. (2008). Visual image reconstruction from human brain activity using a combination of multiscale local image decoders. *Neuron*, *60*(5), 915–929. doi:10.1016/j.neuron.2008.11.004. (Cit. on pp. 13, 27)

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533. doi:10.1038/nature14236. (Cit. on pp. 11, 79, 172)

Moerel, M., Martino, F. D., Uğurbil, K., Yacoub, E. & Formisano, E. (2015). Processing of frequency and location in human subcortical auditory structures. *Scientific Reports*, *5*(1). doi:10.1038/srep17048. (Cit. on p. 112)

Murphy, B., Talukdar, P. P. & Mitchell, T. M. (2012). Selecting corpus-semantic models for neurolinguistic decoding. In E. Agirre, J. Bos & M. T. Diab (Eds.), *Proceedings of the first joint conference on lexical and computational semantics, *sem 2012, june 7-8, 2012, montréal, canada.* (pp. 114–123). Association for Computational Linguistics. (Cit. on pp. 15, 35, 130).

Naselaris, T., Prenger, R. J., Kay, K. N., Oliver, M. & Gallant, J. L. (2009). Bayesian reconstruction of natural images from human brain activity. *Neuron*, *63*(6), 902–915. doi:10.1016/j.neuron.2009.09.006. (Cit. on pp. 10, 27, 28, 40, 57, 150, 155, 165, 166)

Naselaris, T., Kay, K. N., Nishimoto, S. & Gallant, J. L. (2011). Encoding and decoding in fMRI. *NeuroImage*, *56*(2), 400–410. doi:10.1016/j.neuroimage.2010.07.073. (Cit. on pp. 3, 26, 27, 80, 85, 130, 155)

Nishida, S., Gallant, A. G.H.J. L. & Nishimoto, S. (2015). Word statistics in lager-scale texts explain the human cortical semantic representation of objects, actions, and impressions. In *Neuroscience*. (Cit. on pp. 15, 19, 130).

Nishimoto, S., Vu, A. T., Naselaris, T., Benjamini, Y., Yu, B. & Gallant, J. (2014). Gallant lab natural movie 4t fmri data. CRCNS.org. doi:10.6080/k00z715x. (Cit. on pp. 86, 133, 167, 170)

Nishimoto, S., Vu, A. T., Naselaris, T., Benjamini, Y., Yu, B. & Gallant, J. L. (2011a). Reconstructing visual experiences from brain activity evoked by natural movies. *Current Biology*, *21*(19), 1641–1646. doi:10.1016/j.cub.2011.08.031. (Cit. on pp. 10, 27, 40, 103, 133, 134, 138, 167, 170)

Nishimoto, S., Vu, A. T., Naselaris, T., Benjamini, Y., Yu, B. & Gallant, J. L. (2011b). Reconstructing visual experiences from brain activity evoked by natural movies. *Current Biology*, *21*(19), 1641–1646. doi:10.1016/j.cub.2011.08.031. (Cit. on pp. 85, 86, 93, 96)

Norris, D. G. (2006). Principles of magnetic resonance assessment of brain function. *Journal of Magnetic Resonance Imaging*, *23*(6), 794–807. doi:10.1002/jmri.20587. (Cit. on pp. 79, 131)

Oliva, A. & Torralba, A. (2001). *International Journal of Computer Vision*, *42*(3), 145–175. doi:10.1023/a:1011139631724. (Cit. on pp. 12, 135, 137, 170)

Olshausen, B. A. & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*(6583), 607–609. doi:10.1038/381607a0. (Cit. on pp. 27, 79)

Parker, A. J. & Hawken, M. J. (1988). Two-dimensional spatial structure of receptive fields in monkey striate cortex. *Journal of the Optical Society of America A*, *5*(4), 598. doi:10.1364/josaa.5.000598. (Cit. on p. 39)

Pasley, B. N., David, S. V., Mesgarani, N., Flinker, A., Shamma, S. A., Crone, N. E., ... Chang, E. F. (2012). Reconstructing speech from human auditory cortex. *PLoS Biology*, *10*(1), e1001251. doi:10.1371/journal.pbio.1001251. (Cit. on p. 26)

Patterson, K., Nestor, P. J. & Rogers, T. T. (2007). Where do you know what you know? the representation of semantic knowledge in the human brain. *Nature Reviews Neuroscience*, *8*(12), 976–987. doi:10.1038/nrn2277. (Cit. on p. 78)

Patterson, R. D., Uppenkamp, S., Johnsrude, I. S. & Griffiths, T. D. (2002). The processing of temporal pitch and melody information in auditory cortex. *Neuron*, *36*(4), 767–776. doi:10.1016/s0896-6273(02)01060-7. (Cit. on p. 112)

Pedregosa, F., Eickenberg, M., Ciuciu, P., Gramfort, A. & Thirion, B. (2014). Data-driven HRF estimation for encoding and decoding models. *CoRR*, *abs/1402.7015*. (Cit. on p. 80).

Pedregosa, F., Eickenberg, M., Ciuciu, P., Thirion, B. & Gramfort, A. (2015). Data-driven HRF estimation for encoding and decoding models. *NeuroImage*, *104*, 209–220. doi:10.1016/j.neuroimage.2014.09.060. (Cit. on p. 153)

Pennington, J., Socher, R. & Manning, C. D. (2014). Glove: Global vectors for word representation. In A. Moschitti, B. Pang & W. Daelemans (Eds.), *Proceedings of the 2014 conference on empirical methods in natural language processing, EMNLP 2014, october 25-29, 2014, doha, qatar, A meeting of sigdat, a special interest group of the ACL* (pp. 1532–1543). ACL. (Cit. on p. 15).

Poggio, T. & Riesenhuber, M. (1999). *Nature Neuroscience*, *2*(11), 1019–1025. doi:10.1038/14819. (Cit. on p. 12)

Prenger, R., Wu, M. C.-K., David, S. V. & Gallant, J. L. (2004). Nonlinear v1 responses to natural scenes revealed by neural network analysis. *Neural Networks*, *17*(5-6), 663–679. doi:10.1016/j.neunet.2004.03.008. (Cit. on pp. 9, 19)

Quiroga, R. Q., Reddy, L., Kreiman, G., Koch, C. & Fried, I. (2005). Invariant visual representation by single neurons in the human brain. *Nature*, *435*(7045), 1102–1107. doi:10.1038/nature03687. (Cit. on p. 26)

Rainer, G., Lee, H. & Logothetis, N. K. (2004). The effect of learning on the function of monkey extrastriate visual cortex. *PLoS Biology*, *2*(2), e44. doi:10.1371/journal.pbio.0020044. (Cit. on p. 78)

Rolls, E. T. & Milward, T. (2000). A model of invariant object recognition in the visual system: Learning rules, activation functions, lateral inhibition, and information-based performance measures. *Neural Computation*, *12*(11), 2547–2572. doi:10.1162/089976600300014845. (Cit. on p. 12)

Rombouts, J. O., Bohte, S. M., Martinez-Trujillo, J. & Roelfsema, P. R. (2015). A learning rule that explains how rewards teach attention. *Visual Cognition*, *23*(1-2), 179–205. doi:10.1080/13506285.2015.1010462. (Cit. on p. 172)

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533–536. doi:10.1038/323533a0. (Cit. on p. 9)

Rust, N. C. & Movshon, J. A. (2005). In praise of artifice. *Nature Neuroscience*, *8*(12), 1647–1650. doi:10.1038/nn1606. (Cit. on pp. 77, 103)

Sak, H., Senior, A. W. & Beaufays, F. (2014). Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *CoRR*, *abs/1402.1128*. (Cit. on p. 131).

Santoro, R., Moerel, M., Martino, F. D., Goebel, R., Ugurbil, K., Yacoub, E. & Formisano, E. (2014). Encoding of natural sounds at multiple spectral and temporal resolutions in the human auditory cortex. *PLoS Computational Biology*, *10*(1), e1003412. doi:10.1371/journal.pcbi.1003412. (Cit. on p. 112)

Saxe, A. M., Bhand, M., Mudur, R., Suresh, B. & Ng, A. Y. (2011). Unsupervised learning models of primary cortical receptive fields and receptive field plasticity. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira & K. Q. Weinberger (Eds.), *Advances in neural information processing systems 24: 25th annual conference on neural information processing systems 2011. proceedings of a meeting held 12-14 december 2011, granada, spain.* (pp. 1971–1979). (Cit. on p. 52).

Scellier, B. & Bengio, Y. (2017). Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in Computational Neuroscience*, *11*. doi:10.3389/fncom.2017.00024. (Cit. on p. 172)

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117. doi:10.1016/j.neunet.2014.09.003. (Cit. on pp. 6, 84, 88)

Schoenmakers, S., Barth, M., Heskes, T. & van Gerven, M. (2013). Linear reconstruction of perceived images from human brain activity. *NeuroImage*, *83*, 951–961. doi:10.1016/j.neuroimage.2013.07.043. (Cit. on pp. 8, 19, 27)

Schultz, W., Dayan, P. & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593–1599. doi:10.1126/science.275.5306.1593. (Cit. on p. 79)

Schwartz, B. L. & Krantz, J. H. (2015). *Sensation and perception* (1st ed.). SAGE Publications, Inc. (Cit. on p. 112).

Seeliger, K., Fritsche, M., Güçlü, U., Schoenmakers, S., Schoffelen, J.-M., Bosch, S. E. & van Gerven, M. A. J. (2017). Cnn-based encoding and decoding of visual object recognition in space and time. *bioRxiv*. doi:10.1101/118091. (Cit. on pp. 12, 174)

Seibert, D., Yamins, D. L., Ardila, D., Hong, H., DiCarlo, J. J. & Gardner, J. L. (2016). A performance-optimized model of neural responses across the ventral visual stream. *bioRxiv*. doi:10.1101/036475. (Cit. on pp. 12, 113, 126, 174)

Semeniuta, S., Severyn, A. & Barth, E. (2016). Recurrent dropout without memory loss. *CoRR*, *abs/1603.05118*. (Cit. on p. 156).

Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M. & Poggio, T. (2007). Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(3), 411–426. doi:10.1109/tpami.2007.56. (Cit. on p. 80)

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., ... Hassabis, D. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489. doi:10.1038/nature16961. (Cit. on p. 11)

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, *abs/1409.1556*. (Cit. on pp. 62, 104).

Simonyan, K., Vedaldi, A. & Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, *abs/1312.6034*. (Cit. on p. 103).

Smith, A. (2001). Estimating receptive field size from fMRI data in human striate and extrastriate visual cortex. *Cerebral Cortex*, *11*(12), 1182–1190. doi:10.1093/cercor/11.12.1182. (Cit. on pp. 43, 76)

Song, H. F., Yang, G. R. & Wang, X.-J. (2016). Training excitatory-inhibitory recurrent neural networks for cognitive tasks: A simple and flexible framework. *PLOS Computational Biology*, *12*(2), e1004792. doi:10.1371/journal.pcbi.1004792. (Cit. on p. 173)

Soomro, K., Zamir, A. R. & Shah, M. (2012). *Ucf101: A dataset of 101 human actions classes from videos in the wild*. University of Central Florida. (Cit. on p. 90).

Springenberg, J. T., Dosovitskiy, A., Brox, T. & Riedmiller, M. A. (2014). Striving for simplicity: The all convolutional net. *CoRR*, *abs/1412.6806*. (Cit. on p. 103).

Staeren, N., Renvall, H., Martino, F. D., Goebel, R. & Formisano, E. (2009). Sound categories are represented as distributed patterns in the human auditory cortex. *Current Biology*, *19*(6), 498–502. doi:10.1016/j.cub.2009.01.066. (Cit. on p. 112)

Sutskever, I., Martens, J. & Hinton, G. E. (2011). Generating text with recurrent neural networks. In L. Getoor & T. Scheffer (Eds.), *Proceedings of the 28th international conference on machine learning, ICML 2011, bellevue, washington, usa, june 28 - july 2, 2011* (pp. 1017–1024). Omnipress. (Cit. on p. 131).

Sutskever, I., Vinyals, O. & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence & K. Q. Weinberger (Eds.), *Advances in neural information processing systems 27: Annual conference on neural information processing systems 2014, december 8-13 2014, montreal, quebec, canada* (pp. 3104–3112). (Cit. on p. 11).

Swisher, J. D., Gatenby, J. C., Gore, J. C., Wolfe, B. A., Moon, C.-H., Kim, S.-G. & Tong, F. (2010). Multiscale pattern analysis of orientation-selective activity in the primary visual cortex. *Journal of Neuroscience*, *30*(1), 325–330. doi:10.1523/jneurosci.4811-09.2010. (Cit. on p. 43)

Tanaka, K. (1996). Inferotemporal cortex and object vision. *Annual Review of Neuroscience*, *19*(1), 109–139. doi:10.1146/annurev.ne.19.030196.000545. (Cit. on p. 56)

Tanigawa, H., Lu, H. D. & Roe, A. W. (2010). Functional organization for color and orientation in macaque v4. *Nature Neuroscience*, *13*(12), 1542–1548. doi:10.1038/nn.2676. (Cit. on p. 78)

Thirion, B., Duchesnay, E., Hubbard, E., Dubois, J., Poline, J.-B., Lebihan, D. & Dehaene, S. (2006). Inverse retinotopy: Inferring the visual content of images from brain activation patterns. *NeuroImage*, *33*(4), 1104–1116. doi:10.1016/j.neuroimage.2006.06.062. (Cit. on pp. 13, 27, 78)

Toiviainen, P., Alluri, V., Brattico, E., Wallentin, M. & Vuust, P. (2014). Capturing the musical brain with lasso: Dynamic decoding of musical features from fMRI data. *NeuroImage*, *88*, 170–180. doi:10.1016/j.neuroimage.2013.11.017. (Cit. on pp. 112, 113)

Tootell, R. B., Silverman, M. S., Hamilton, S. L., Switkes, E. & Valois, R. L. D. (1988). Functional anatomy of macaque striate cortex. v. spatial frequency. *Journal of Neuroscience*, *8*(5), 1610–1624. (Cit. on p. 41).

Tran, D., Bourdev, L. D., Fergus, R., Torresani, L. & Paluri, M. (2014). C3D: generic features for video analysis. *CoRR, abs/1412.0767*. (Cit. on pp. 88, 89, 91, 168).

Valois, R. L. D., Albrecht, D. G. & Thorell, L. G. (1982). Spatial frequency selectivity of cells in macaque visual cortex. *Vision Research*, *22*(5), 545–559. doi:10.1016/0042-6989(82)90113-4. (Cit. on p. 39)

van Gerven, M. A. J., de Lange, F. P. & Heskes, T. (2010). Neural decoding with hierarchical generative models. *Neural Computation*, *22*(12), 3127–3142. doi:10.1162/neco_a_00047. (Cit. on pp. 13, 16, 19, 53, 56, 174)

van den Oord, A., Dieleman, S. & Schrauwen, B. (2014). Transfer learning by supervised pre-training for audio-based music classification. In H. Wang, Y. Yang & J. H. Lee (Eds.), *Proceedings of the 15th international society for music information retrieval conference, ISMIR 2014, taipei, taiwan, october 27-31, 2014* (pp. 29–34). (Cit. on p. 120).

Vu, V. Q., Ravikumar, P., Naselaris, T., Kay, K. N., Gallant, J. L. & Yu, B. (2011). Encoding and decoding v1 fMRI responses to natural images with sparse nonparametric models. *The Annals of Applied Statistics*, *5*(2B), 1159–1182. doi:10.1214/11-aoas476. (Cit. on pp. 27, 45, 157)

Werbos, P. (1990). Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, *78*(10), 1550–1560. doi:10.1109/5.58337. (Cit. on p. 18)

Widrow, B. & Hoff, M. E. (1960). *Adaptive switching circuits*. Stanford University. (Cit. on p. 8).

Wray, J. & Green, G. G. R. (1994). Calculation of the volterra kernels of non-linear dynamic systems using an artificial neural network. *Biological Cybernetics*, *71*(3), 187–195. doi:10.1007/bf00202758. (Cit. on p. 132)

Wu, M. C.-K., David, S. V. & Gallant, J. L. (2006). Complete functional characterization of sensory neurons by system identification. *Annual Review of Neuroscience*, *29*(1), 477–505. doi:10.1146/annurev.neuro.29.051605.113024. (Cit. on p. 78)

Yacoub, E., Harel, N. & Ugurbil, K. (2008). High-field fMRI unveils orientation columns in humans. *Proceedings of the National Academy of Sciences*, *105*(30), 10607–10612. doi:10.1073/pnas.0804110105. (Cit. on p. 53)

Yamada, K., Miyawaki, Y. & Kamitani, Y. (2015). Inter-subject neural code converter for visual image representation. *NeuroImage*, *113*, 289–297. doi:10.1016/j.neuroimage.2015.03.059. (Cit. on pp. 84, 105)

Yamins, D. L. K., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D. & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, *111*(23), 8619–8624. doi:10.1073/pnas.1403112111. (Cit. on pp. 11, 19, 52, 80, 103, 113, 130, 174)

Yamins, D. L. K. & DiCarlo, J. J. (2016a). Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, *19*(3), 356–365. doi:10.1038/nn.4244. (Cit. on p. 156)

Yamins, D. L. & DiCarlo, J. J. (2016b). Eight open questions in the computational modeling of higher sensory cortex. *Current Opinion in Neurobiology*, *37*, 114–120. doi:10.1016/j.conb.2016.02.001. (Cit. on p. 132)

Zaremba, W., Sutskever, I. & Vinyals, O. (2014). Recurrent neural network regularization. *CoRR*, *abs/1409.2329*. (Cit. on p. 156).

Zeiler, M. D. & Fergus, R. (2013). Visualizing and understanding convolutional networks. *CoRR*, *abs/1311.2901*. (Cit. on pp. 11, 56, 63, 103).

# Nederlandse samenvatting

In de afgelopen jaren zijn encoding (generatieve) en decoding (classificatie/reconstructie) modellen steeds populairder geworden als onderzoeksmethoden om de relatie tussen sensorische stimuli en de corresponderende hersenactiviteit (gemeten met bijvoorbeeld functionele MRI) te karakteriseren. Bij conventionele methoden worden de verschillende eigenschappen van de stimuli (bijvoorbeeld oriëntatie of semantische betekenis) die moeten worden verklaard door de modellen, handmatig toegekend door de onderzoekers. Dit proces is moeilijk en tijdrovend, en bovendien erg afhankelijk van de persoonlijke verwachtingen en hypotheses van de onderzoeker. Mede hierdoor is het nog altijd onduidelijk welke eigenschappen van stimuli waar en hoe worden gerepresenteerd in het brein.

Om deze onduidelijkheden te overkomen, introduceerden we in hoofdstuk 2 een algemeen framework, een zogenaamd diep neuraal netwerk, dat testbare voorspellingen kan doen over hoe de activiteit van een voxel zich verhoudt tot statistisch gevormde representaties van natuurlijke stimuli (zoals plaatjes of filmpjes). De resultaten van dit hoofdstuk toonden aan dat: i) de statistische representaties van plaatjes, gemaakt door ons model, lijken op de representaties in de vroege visuele cortex. Onze geleerde statistische representaties gelijken zelfs beter dan de handgemaakte representaties uit conventionele methoden. ii) in tegenstelling tot handgemaakte representaties, die alleen verklarend zijn voor activiteit in de vroege visuele cortex, kan ons model ook de representaties van hersengebieden hoger in de visuele hiërarchie verklaren.

In Hoofdstuk 3 bouwden we voort op de bevindingen in Hoofdstuk 2. We gebruikten het diep neuraal netwerk om de hogere en complexere visuele representaties in de ventrale visuele cortex te bestuderen. We vonden: i) een gelijkende representatieve gradiënt in het brein en in de lagen van het netwerk: de representaties van diepere lagen in het neurale netwerk komen beter overeen met de neurale representaties in hogere visuele gebieden. ii) dat deze overeenkomst onafhankelijk is van de keuze voor de architectuur van het neurale netwerk.

In Hoofdstuk 4 hebben we de ideeën uit het vorige hoofdstuk toegepast op actieherkenning in de dorsale visuele stroom. De belangrijkste resultaten van dit hoofdstuk waren dat: i) representaties in het diep neurale netwerk in de dorsale stroom vertonen gelijkenissen. ii) representaties in de dorsale visuele stroom overeenkomen tussen proefpersonen. iii) een algemeen encoding model, getraind op data van verschillende proefpersonen, fMRI-responsen kan voorspellen op nieuwe stimuli voor nieuwe proefpersonen. iv) een algemeen decoding model, getraind op data van verschillende proefpersonen, nieuwe stimuli kan identificeren uit fMRI-responsen van nieuwe proefpersonen.

In Hoofdstuk 5 verlegden we de focus van visuele naar auditieve representaties. In dit hoofdstuk bestudeerden we of de overeenkomst tussen representaties in het neurale netwerk en het brein ook bestaat het auditieve systeem. De belangrijkste resultaten van dit hoofdstuk waren dat: i) neurale representaties in hogere auditieve gebieden in de superior temporal gyrus (STG) corresponderen met representaties in diepere geoptimaliseerde lagen van het auditieve neurale netwerk. ii) bij het modelleren van neurale representaties in STG is het

belangrijk het diep neurale netwerk te optimaliseren voor de stimuli die moeten worden gerepresenteerd. iii) onze resultaten overeenkomen met de literatuur in de visuele hersenwetenschappen. De resultaten uit Hoofdstukken 3 en 5 suggereren het bestaan van verschillende representatieve gradiënten in de sensorische stromen van het menselijk brein.

In Hoofdstuk 6 gebruikten we een recurrente neurale netwerk (RNN) als een encoding model voor het voorspellen van de relatie tussen sensorische stimulus en hersenrespons. We vergeleken de RNNs met verschillende lineaire convolutiemodellen. We definieerden twee representatieniveaus: één om vroeg-sensorische responsen en één om complexere, semantische responsen te voorspellen. We analyseerden dezelfde dataset als die in Hoofdstuk 4, waar participanten een film keken in de MRI-scanner. In dit hoofdstuk was de benadering echter anders dan die in Hoofdstukken 2-5, waarin we de neurale netwerken uitsluitend gebruikten om de transformatie van sensorische stimuli naar hersenrepresentatie te modelleren. We vonden dat: i) een RNN niet alleen de transformatie van stimulus naar hersenrepresentatie kan leren (en dat beter doet dan lineaire convolutiemodellen), maar ook de temporele afhankelijkheden van de hersenresponsen. ii) een RNN hemodynamische responsfuncties kan modelleren in fMRI-experimenten. iii) de geleerde semantische representaties in het netwerk kunnen hersenrepresentaties van semantische inhoud van waargenomen natuurlijke films voorspellen.

Sinds de eerste modellen voor het representeren van sensorische informatie van ruim 50 jaar geleden verschenen, zijn er steeds geavanceerdere methoden ontwikkeld om te begrijpen hoe ons brein de inkomende informatie van de zintuigen representeert. In deze thesis hebben we aangetoond dat kunstmatige neurale netwerken de volgende stap zijn in deze evolutie: neurale netwerken verklaren de hersenresponsen beter dan bestaande alternatieve modellen. De netwerken verklaren niet alleen de responsen op een algoritmisch niveau (i.e. de functies van hersengebieden in het representeren van de wereld), maar ze kunnen ook inzicht bieden in de manier waarop het brein die informatie representeert (het implementatieniveau). Gegeven de recente ontwikkelingen met betrekking tot biologisch plausibele leermechanismen en netwerkarchitecturen, verwachten we dat neurale netwerken belangrijke inzichten kunnen brengen over zowel de functies als de mechanismen van de representaties in de hersenen. Door verschillende van zulke generatieve modellen van neurale data te vergelijken, kunnen we steeds preciezer modelleren hoe het brein werkt.

# Acknowledgements

After four years of intense work, my doctoral thesis is finally complete. Here is a shout-out to the people who made it possible.

First and foremost, I would like to thank my supervisors Marcel, Peter and David. In particular, Marcel has been the best supervisor that I could have ever asked for. Words cannot do justice to everything that he has done for me, and I am thankful for it all. I have been very lucky to have him as a colleague, a mentor and a friend.

My thanks also go to: My collaborators Eric, Hugo, Isabelle, Jordi, Julia, Luca, Lyuben, Meysam, Michael, Rob, Sergio and Xavier, labmates Ali, Andrew, Claudia, Gabi, Elena, Erdi, Haiteng, Irina, Katja, Linda, Marieke, Max, Nadine, Pasi, Ronald, Sander, Sanne and Silvan, and students Diede, Jordi, Jordy, Kevin, Max, Rowan, Simon and Thomas for sciencing with me (and another one to Luca for indulging in a daily dose of brainstorming with me); my colleagues Jose and Paul for showing me the ropes of Prisma, Skyra and Trio (and to Prisma, Skyra and Trio for showing me the brains of people), and Antonette, Helma, Jolanda, Lieve, Maaike and Vanessa for being invariably helpful (and another one to Jolanda for keeping me in the loop about the life on the outside with her very insightful emails); my relatives Begüm, Caner, Cüneyt, Ergin, Erol, Güçlü, Günay, Hamide, Mustafa, Nehir, Yalın and Yurdanur, and other aunts, cousins and uncles for giving me their unconditional support (and another one to Ergin for literally setting the wheels in motion and Erol for figuratively setting the wheels in motion).

Last but not least, I would like to thank my family Kazim, Mine and Özlem (pingback), and significant other Yağmur for they know what.

# Curriculum vitae

## Experience

- **Assistant Professor**
  Radboud University, Nijmegen, Netherlands, 2018 –
- **Postdoctoral Researcher**
  Radboud University, Nijmegen, Netherlands, 2017 – 2018
- **Doctoral Researcher**
  Radboud University, Nijmegen, Netherlands, 2013 – 2017
- **Research Assistant**
  Radboud University, Nijmegen, Netherlands, 2012

## Education

- **Ph.D. Cognitive Neuroscience**
  Radboud University, Nijmegen, Netherlands, 2013 – 2017
  Supervisor: Prof. Marcel van Gerven; Thesis: Neural coding with deep learning
- **M.Sc. Cognitive Neuroscience**
  Radboud University, Nijmegen, Netherlands, 2011 – 2013
  Grade: 8.80/10.00; Honors: Cum laude; Supervisor: Prof. Marcel van Gerven; Thesis: Unsupervised learning of linearizing feature spaces for encoding and decoding in functional magnetic resonance imaging
- **B.IT Artificial Intelligence**
  Multimedia University, Malacca, Malaysia, 2008 – 2011
  Grade: 3.95/4.00; Honors: First-class; Supervisor: Prof. Andrews Samraj; Thesis: A spatiotemporal analysis of electroencephalogram signals

## Awards and fellowships

- **University Study Award** (master's award)
  Radboud University, Nijmegen, Netherlands, 2014
- **TOPtalent Competition** (four-year doctoral fellowship)
  Radboud University, Nijmegen, Netherlands, 2013
- **Academy Assistants Programme** (one-year research assistant fellowship)
  Royal Netherlands Academy of Arts and Sciences, Amsterdam, Netherlands, 2012
- **Huygens Scholarship Programme** (two-year master's fellowship)
  Netherlands Universities Foundation for International Cooperation, The Hague, Netherlands, 2011
- **Book Award** (bachelor's award)
  Multimedia University, Malacca, Malaysia, 2011

## Publications and presentations

### Preprint

- Jacques Junior, J., Güçlütürk, Y., Pérez, M., **Güçlü, U.**, Andujar, C., Baró, X., Escalante, H., Guyon, I., van Gerven, M., van Lier, R., Escalera, S. (2018). First impressions: a survey on computer vision-based apparent personality trait analysis. ***arXiv preprint arXiv:1804.08046 [cs.CV]***. https://arxiv.org/pdf/1804.08046.pdf (https://arxiv.org/pdf/1804.08046.pdf)
- Escalante, H., Kaya, H., Salah, A., Escalera, S., Güçlütürk, Y., **Güçlü, U.**, Baró, X., Guyon, I., Jacques Junior, J., Madadi, M., Ayache, S., Viegas, E., Gürpınar, F., Wicaksana, A., Liem, C., van Gerven, M., and van Lier, R. (2018). Explaining first impressions: modeling, recognizing, and explaining apparent personality from videos. ***arXiv preprint arXiv:1802.00745 [cs.CV]***. https://arxiv.org/pdf/1802.00745.pdf (https://arxiv.org/pdf/1802.00745.pdf)
- Seeliger, K., **Güçlü, U.**, Ambrogioni, L., Güçlütürk, Y., and van Gerven, M. (2017). Generative adversarial networks for reconstructing natural images from brain activity. ***bioRxiv***. https://doi.org/10.1101/226688 (https://doi.org/10.1101/226688)
- Ambrogioni, L., **Güçlü, U.**, van Gerven, M., and Maris, E. (2017). The kernel mixture network: a nonparametric method for conditional density estimation of continuous random variables. ***arXiv preprint arXiv:1705.07111 [stat.ML]***. https://arxiv.org/pdf/1705.07111.pdf (https://arxiv.org/pdf/1705.07111.pdf)
- **Güçlü, U.***, Güçlütürk, Y.*, Madadi, M., Escalera, S., Baró, X., González, J., van Lier, R., and van Gerven, M. (2017). End-to-end semantic face

segmentation with conditional random fields as convolutional, recurrent and adversarial networks. ***arXiv preprint arXiv:1703.03305 [cs.CV]***. https://arxiv.org/pdf/1703.03305.pdf (https://arxiv.org/pdf/1703.03305.pdf)

- Ambrogioni, L., **Güçlü, U.**, Maris, E., and van Gerven, M. (2017). Estimating nonlinear dynamics with the convnet smoother. ***arXiv preprint arXiv:1702.05243 [stat.ML]***. https://arxiv.org/pdf/1702.05243.pdf (https://arxiv.org/pdf/1702.05243.pdf)

## Journal

- Güçlütürk, Y., **Güçlü, U.**, van Gerven, M., and van Lier, R. (2018). Representations of naturalistic stimulus complexity in early and associative visual and auditory cortices. ***Scientific Reports***, 8:3439. https://doi.org/10.1038/s41598-018-21636-y (https://doi.org/10.1038/s41598-018-21636-y)
- Güçlütürk, Y., **Güçlü, U.**, Baró, X., Escalante, H., Guyon, I., Escalera, S., van Gerven, M., and van Lier, R. (2017). Multimodal first impression analysis with deep residual networks. ***IEEE Transactions on Affective Computing***. https://doi.org/10.1109/TAFFC.2017.2751469 (https://doi.org/10.1109/TAFFC.2017.2751469)
- Berezutskaya, J., Freudenburg, Z., **Güçlü, U.**, van Gerven, M., and Ramsey, N. (2017). Neural tuning to low-level features of speech throughout the perisylvian cortex. ***The Journal of Neuroscience***, 37(33):7906–7920. https://doi.org/10.1523/JNEUROSCI.0238-17.2017 (https://doi.org/10.1523/JNEUROSCI.0238-17.2017)
- Seeliger, K., Fritsche, M., **Güçlü, U.**, Schoenmakers, S., Schoffelen, J.-M., Bosch, S., and van Gerven, M. (2017). Convolutional neural network-based encoding and decoding of visual object recognition in space and time. ***NeuroImage***. https://doi.org/10.1016/j.neuroimage.2017.07.018 (https://doi.org/10.1016/j.neuroimage.2017.07.018)
- **Güçlü, U.** and van Gerven, M. (2017). Modeling the dynamics of human brain activity with recurrent neural networks. ***Frontiers in Computational Neuroscience***, 11:7. https://doi.org/10.3389/fncom.2017.00007 (https://doi.org/10.3389/fncom.2017.00007)
- **Güçlü, U.** and van Gerven, M. (2015). Increasingly complex representations of natural movies across the dorsal stream are shared between subjects. ***NeuroImage***, 145(Part B):329–336. https://doi.org/10.1016/j.neuroimage.2015.12.036 (https://doi.org/10.1016/j.neuroimage.2015.12.036)
- **Güçlü, U.** and van Gerven, M. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. ***The Journal of Neuroscience***, 35(27):10005–10014. https://doi.org/10.1523/JNEUROSCI.5023-14.2015 (https://doi.org/10.1523/JNEUROSCI.5023-14.2015)
- Schoenmakers, S., **Güçlü, U.**, van Gerven, M., and Heskes, T. (2015). Gaussian mixture models and semantic gating improve reconstructions from human brain activity. ***Frontiers in Computational Neuroscience***, 8:173. https://doi.org/10.3389/fncom.2014.00173 (https://doi.org/10.3389/fncom.2014.00173)
- **Güçlü, U.** and van Gerven, M. (2014). Unsupervised feature learning improves prediction of human brain activity in response to natural images. ***PLOS Computational Biology***, 10(8):e1003724. https://doi.org/10.1371/journal.pcbi.1003724 (https://doi.org/10.1371/journal.pcbi.1003724) (best thesis award by Radboud University; best poster award by Donders Institute)

## Conference

### Paper

- Ambrogioni, L., Berezutskaya, J., **Güçlü, U.**, van den Borne, E., Güçlütürk, Y., van Gerven, M., and Maris, E. (2017). Bayesian model ensembling using meta-trained recurrent neural networks. In ***Workshop on Meta-Learning - Neural Information Processing Systems***. http://metalearning.ml/papers/metalearn17_ambrogioni.pdf (http://metalearning.ml/papers/metalearn17_ambrogioni.pdf) ‡
- **Güçlü, U.***, Güçlütürk, Y.*, Ambrogioni, L., Maris, E., van Lier, R., and van Gerven, M. (2017). Algorithmic composition of polyphonic music with the wavecrf. In ***Machine Learning for Creativity and Design - Neural Information Processing Systems***. https://nips2017creativity.github.io/doc/WaveCRF.pdf (https://nips2017creativity.github.io/doc/WaveCRF.pdf) ‡
- Güçlütürk, Y.*, **Güçlü, U.***, Seeliger, K., Bosch, S., van Lier, R., and van Gerven, M. (2017). Reconstructing perceived faces from brain activations with deep adversarial neural decoding. In ***Neural Information Processing Systems***. https://papers.nips.cc/paper/7012-reconstructing-perceived-faces-from-brain-activations-with-deep-adversarial-neural-decoding.pdf (https://papers.nips.cc/paper/7012-reconstructing-perceived-faces-from-brain-activations-with-deep-adversarial-neural-decoding.pdf) ‡ (best poster award by Donders Institute)
- Güçlütürk, Y., **Güçlü, U.**, Pérez, M., Escalante, H., Baró, X., Guyon, I., Andujar, C., Jacques Junior, J., Madadi, M., Escalera, S., van Gerven, M., and van Lier, R. (2017). Visualizing apparent personality analysis with deep residual networks. In ***Action, Gesture, and Emotion Recognition Workshop and Competitions: Large Scale Multimodal Gesture Recognition and Real versus Fake Expressed Emotions - International Conference on Computer Vision***. http://openaccess.thecvf.com/content_ICCV_2017_workshops/papers/w44/Gucluturk_Visualizing_Apparent_Personality_ICCV_2017_paper.pdf (http://openaccess.thecvf.com/content_ICCV_2017_workshops/papers/w44/Gucluturk_Visualizing_Apparent_Personality_ICCV_2017_paper.pdf) ‡
- Berezutskaya, J., Freudenburg, Z., Ramsey, N., **Güçlü, U.**, and van Gerven, M. (2017). Modeling brain responses to perceived speech with lstm networks. In ***Benelux Conference on Machine Learning***. https://pure.tue.nl/ws/files/72619856/benelearn_2017.pdf#page=10 (https://pure.tue.nl/ws/files/72619856/benelearn_2017.pdf#page=10) †
- Escalante, H., Guyon, I., Escalera, S., Jacques Junior, J., Madadi, M., Baró, X., Ayache, S., Viegas, E., Güçlütürk, Y., **Güçlü, U.**, et al. (2017). Design of an explainable machine learning challenge for video interviews. In ***International Joint Conference on Neural Networks***. https://doi.org/10.1109/IJCNN.2017.7966320 (https://doi.org/10.1109/IJCNN.2017.7966320) †
- **Güçlü, U.**, Thielen, J., Hanke, M., and van Gerven, M. (2016). Brains on beats. In ***Neural Information Processing Systems***. https://papers.nips.cc/paper/6222-brains-on-beats.pdf (https://papers.nips.cc/paper/6222-brains-on-beats.pdf) ‡
- Güçlütürk, Y., **Güçlü, U.**, van Gerven, M., and van Lier, R. (2016). Deep impression: audiovisual deep residual networks for multimodal

apparent personality trait recognition. In *Challenge on Automatic Personality Analysis - European Conference on Computer Vision*. https://doi.org/10.1007/978-3-319-49409-8_28 (https://doi.org/10.1007/978-3-319-49409-8_28) †

- Güçlütürk, Y.*, **Güçlü, U.***, van Lier, R., and van Gerven, M. (2016). Convolutional sketch inversion. In *Computer Vision for Art Analysis - European Conference on Computer Vision*. https://doi.org/10.1007/978-3-319-46604-0_56 (https://doi.org/10.1007/978-3-319-46604-0_56) †
- **Güçlü, U.** and van Gerven, M. (2015). Semantic vector space models predict neural responses to complex visual stimuli. In *Machine Learning and Interpretation in Neuroimaging - Neural Information Processing Systems*. https://arxiv.org/pdf/1510.04738.pdf (https://arxiv.org/pdf/1510.04738.pdf) ‡
- **Güçlü, U.** and van Gerven, M. (2013). Unsupervised learning of features for bayesian decoding in functional magnetic resonance imaging. In *Benelux Conference on Machine Learning*. http://benelearn2013.org/pdfs/paper_27.pdf (http://benelearn2013.org/pdfs/paper_27.pdf) ‡

**Demo**

- Escalera, S., Guyon, I., Chen, B., Quintana, M., **Güçlü, U.**, Güçlütürk, Y., Baró, X., van Lier, R., Andujar, C., van Gerven, M., Boser, B., and Wang, L. (2016). Biometric applications of CNNs: get a job at "Impending Technologies"!. In *Neural Information Processing Systems*. ‡

**Abstract**

- **Güçlü, U.**, Güçlütürk, Y., Ambrogioni, L., Maris, E., van Lier, R., and van Gerven, M. (2017). Ultrafast HRF and pRF estimation with deep learning. In *De Nederlandse Vereniging voor Psychonomie*. ‡
- Seeliger, K., Fritsche, M., **Güçlü, U.**, Schoenmakers, S., Schoffelen, J.-M., Bosch, S., and van Gerven, M. (2016). A forward pass through the visual system: convnets encode MEG source activity", In *Representation Learning in Artificial and Biological Neural Networks - Neural Information Processing Systems*. ‡
- Berezutskaya, J., Freudenburg, Z., **Güçlü, U.**, van Gerven, M., and Ramsey, N. (2016). Neural tuning to low-level features of complex sound in posterior superior temporal gyrus and beyond. In *Society for Neuroscience*. ‡
- Güçlütürk, Y., **Güçlü, U.**, Jacobs, R., van Gerven, M., and van Lier, R. (2016). Neural correlates of liking and perceived complexity of music. In *International Association of Empirical Aesthetics*. ‡
- **Güçlü, U.**, and van Gerven, M. (2016). Neural encoding of faces with deep neural networks. In *Organization for Human Brain Mapping*. ‡
- Seeliger, K., Fritsche, M., **Güçlü, U.**, Schoenmakers, S., Bosch, S., and van Gerven, M. (2016). Convolutional neural networks code for spatiotemporal MEG source activity across the visual system, In *ICT.Open*. ‡
- **Güçlü, U.**, and van Gerven, M. (2015). Semantic vector space models predict neural responses to complex visual stimuli. In *De Nederlandse Vereniging voor Psychonomie*. ‡
- **Güçlü, U.**, Knechten, M., and van Gerven, M. (2014). A two-stage approach to estimating voxel-specific encoding models improves prediction of hemodynamic responses to natural images. In *International Neuroinformatics Coordinating Facility*. ‡
- van Lier, R., **Güçlü, U.**, and Koning, A. (2013). Appreciation of afterimages in contemporary art: an eye movement study. In *European Conference on Visual Perception* . ‡
- **Güçlü, U.**, and van Gerven, M. (2013). Unsupervised learning of invariant features for encoding fMRI responses to natural images. In *Organization for Human Brain Mapping*. ‡

## Chapter

- **Güçlü, U.** and van Gerven, M. (2017). Probing human brain function with artificial neural networks. In *Computational Models of Brain and Behavior (ed. A. Moustafa)*, pages 413–423. John Wiley & Sons. https://doi.org/10.1002/9781119159193.ch30 (https://doi.org/10.1002/9781119159193.ch30)

*equal contribution; †oral presentation; ‡poster presentation

# Donders Graduate School for Cognitive Neuroscience

For a successful research Institute, it is vital to train the next generation of young scientists. To achieve this goal, the Donders Institute for Brain, Cognition and Behaviour established the Donders Graduate School for Cognitive Neuroscience (DGCN), which was officially recognised as a national graduate school in 2009. The Graduate School covers training at both Master's and PhD level and provides an excellent educational context fully aligned with the research programme of the Donders Institute.

The school successfully attracts highly talented national and international students in biology, physics, psycholinguistics, psychology, behavioral science, medicine and related disciplines. Selective admission and assessment centers guarantee the enrolment of the best and most motivated students.

The DGCN tracks the career of PhD graduates carefully. More than 50% of PhD alumni show a continuation in academia with postdoc positions at top institutes worldwide, e.g. Stanford University, University of Oxford, University of Cambridge, UCL London, MPI Leipzig, Hanyang University in South Korea, NTNU Norway, University of Illinois, North Western University, Northeastern University in Boston, ETH Zürich, University of Vienna etc.. Positions outside academia spread among the following sectors: specialists in a medical environment, mainly in genetics, geriatrics, psychiatry and neurology. Specialists in a psychological environment, e.g. as specialist in neuropsychology, psychological diagnostics or therapy. Positions in higher education as coordinators or lecturers. A smaller percentage enters business as research consultants, analysts or head of research and development. Fewer graduates stay in a research environment as lab coordinators, technical support or policy advisors. Upcoming possibilities are positions in the IT sector and management position in pharmaceutical industry. In general, the PhDs graduates almost invariably continue with high-quality positions that play an important role in our knowledge economy.

For more information on the DGCN as well as past and upcoming defenses please visit: http://www.ru.nl/donders/graduate-school/phd/ (http://www.ru.nl/donders/graduate-school/phd/)